Dataset Mismatched Steganalysis Using Subdomain Adaptation With Guiding Feature

Lei Zhang
Sichuan University
Sani M. Abdullahi
China Three Gorges University
Peisong He
Sichuan University
Hongxia Wang (hxwang@scu.edu.cn)
Sichuan University - Wangjiang Campus: Sichuan University https://orcid.org/0000-0001-6294-3272

Research Article

Keywords: Image steganalysis, Subdomain adaptation, Cover source mismatch, Steganography

Posted Date: February 15th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1326468/v1

Dataset Mismatched Steganalysis Using Subdomain Adaptation with Guiding Feature

Lei Zhang[1] · Sani M. Abdullahi[2] · Peisong He[1]· Hongxia Wang[1]

Abstract

The generalization problem in deep learning has always been an important problem to be solved. In the field of steganalysis, generalization is also an important factor that makes steganalysis models difficult to deploy in real-world scenarios. For a group of suspicious images that never appeared in the training set, the pre-trained deep learning-based steganalysis models tend to suffer from distinct performance degradation. To address this limitation, in this paper, a feature-guided subdomain adaptation steganalysis framework is proposed to improve the performance of the pre-trained models when detecting new data. Initially, the source domain and target domain will be divided into subdomains according to class, and the distributions of the relevant subdomains are aligned by subdomain adaptation. Afterward, the guiding feature is generated to make the division of subdomains more stable and precise. When it is used to detect three spatial steganographic algorithms with a wide variety of datasets and payloads, the experimental results show that the proposed steganalysis framework can significantly improve the average accuracy of SRNet model by 5.4% at 0.4bpp, 8.5% at 0.2bpp, and 8.0% at 0.1bpp in the case of dataset mismatch.

Keywords Image steganalysis · Subdomain adaptation · Cover source mismatch · Steganography

1 Introduction

Steganography is a kind of covert communication that can leverage the visual redundancy of digital images to embed secret information. The minor changes introduced by steganography are visually imperceptible and the security against steganalysis is the first consideration for steganography. In recent years, many spatial adaptive steganography algorithms have been proposed, such as S-UNIWARD [1], WOW [2], HUGO [3], etc. By analyzing image information, these algorithms can adaptively search for regions with more complex textures in images to minimize the influence of embedding secret information, which brings great challenges to statistical-based steganalysis.

As the opposite of steganography, the purpose of steganalysis is to detect the presence of hidden communication and distinguish between cover and stego images. For traditional steganalysis methods, feature extraction and classifier training are two independent steps. The hand-crafted features used in steganalysis mainly depend on accumulated experience and rich statistical knowledge of steganography. In the past few years, Convolutional Neural Network (CNN) has been adopted to construct powerful steganalysis methods [4-7] to take place of traditional hand-crafted feature-based methods [8-12]. The aforementioned deep learning-based methods [4-7] have more or less introduced the component of hand design. SRNet [13] is the first end-to-end residual structure steganalysis network, which does not adopt hand-designed initialized parameters, and has a good detection accuracy performance of image steganography in both spatial domain and JPEG domain.

However, applying steganalysis tools to real-world scenarios is still very challenging, where testing samples always undergo unknown capturing processes. Generally, the detection error increases when a steganalysis detector trained on one cover source is applied to images from a different source due to the mismatch between both sources. This situation is recognized as the so-called Cover Source Mismatch (CSM). In [14], Fridrich et al. summarized several types of CSM, including payload mismatch, quantization table mismatch, steganographic methods mismatch, mismatched image content, etc. In [15], five mismatch scenarios were

✉ Hongxia Wang
hxwang@scu.edu.cn

✉ Peisong He
gokeyhps@scu.edu.cn

1 School of Cyber Science and Engineering, Sichuan University, Chengdu, 610065, China

2 College of Computer and Information Technology, China Three Gorges University, Yichang 443002, China.

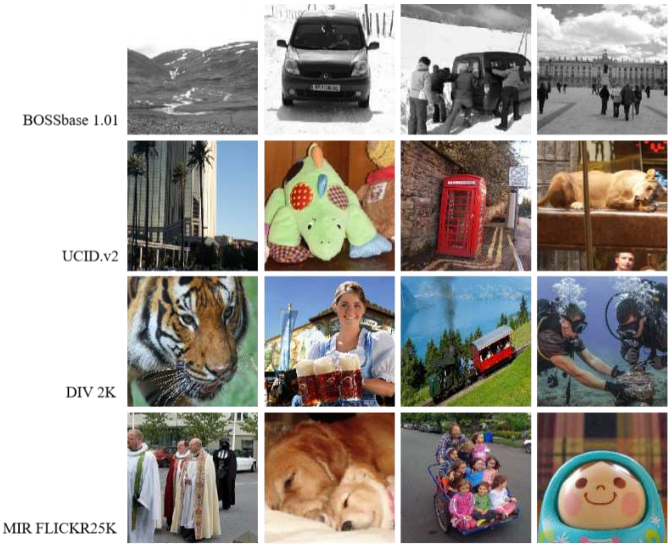


Fig. 1 Differences among the four datasets (four images were randomly selected from each dataset as representatives)

carried out to explore the different effects of these factors on steganalysis, which contain the camera sensors, ISO, image processing pipeline, Quality Factor (QF), and semantic content.

In this paper, dataset mismatch is mainly discussed since we believe that it is more common in realistic application scenarios. In practical applications, the analyzers always need to deal with the collected images from the Internet. However, these suspicious images are more likely to have quite different capturing and processing history compared with the samples used to train the detection model, which can lead to the trained model suffering from a distinct performance drop. More specifically, there are many differences in the images uploaded by various users, including image capturing processes, image content, texture complexity and so on. To illustrate above mentioned difference more intuitively, several datasets are presented in Fig. 1. In order to reduce the effect of dataset mismatch, we take the image intercepted as the target domain, and adjust the parameters of the pre-trained steganalysis model by minimizing the discrepancy between the source domain and the target domain, making the model adapt from the source domain to the target domain. Unlike the supervised steganalysis methods, we believe that test data can be used to train models for adaptation, which is important information for steganalysis.

To date, many works [16-24] have attempted to solve the problem of CSM. We observed that few works are focusing on the CSM problems in relation to deep learning-based steganalysis which occupies a very important part of steganalysis. Most of them studied the mismatch problem on hand-crafted features. Moreover, existing solutions for mismatch problems mostly leveraged a global strategy to align source and target distributions, which cannot effectively describe the boundary between two different classes. In this paper, considering gaps in deep learning-based steganalysis application scenarios and weaknesses of the global alignment strategy, a feature-guided subdomain adaptation steganalysis framework is proposed to improve the performance of a deep learning-based steganalysis model in the case of dataset mismatch. By fusing guiding features, more robust pseudo labels are obtained to divide the subdomains, further stimulating the model to transfer in the right direction, and effectively resisting the influence of the wrong prediction.

In summary, the main contributions of our work can be described as follows.

1) We propose a feature-guided subdomain adaptation steganalysis framework for dataset mismatched detection, which can reduce the distance in feature distributions between subdomains and help improve the performance of deep learning-based steganalysis model in the case of dataset mismatch.

2) We not only use a local alignment strategy to mitigate the undesired variation of related subdomains' distributions but also design a feature-guided module to make subdomain division more precise, which further improves the detection accuracy.

3) By considering various datasets and steganographic methods, we perform extensive experiments to demonstrate the utility of the proposed steganalysis framework.

The rest of this paper is organized as follows. In Section 2, we introduce the related work. In Section 3, our motivation is expounded. In Section 4, our framework is given in detail.

In Section 5, experiments are designed to verify the effectiveness and utility of the proposed framework. Finally, in Section 6, we summarize our work.

2 Related work

For the traditional supervised learning framework, it is hard to construct a reliable steganalysis model with good generalization ability since there is a high cost of collecting a large number of images with various capturing processes. Even if applying data augmentation in the training phase can improve the generalization performance of steganalysis models, the performance gain is still limited [25]. Therefore, cover source mismatch is still an important issue to be studied in the field of steganalysis.

In recent years, several methods have been proposed to mitigate the negative effects of cover source mismatch in steganalysis. Generally, they can be divided into two categories, which are subspace method and classifier construction method.

For the first kind of method, researchers tried to find a projection to map the features into a common subspace to reduce the distribution discrepancy between the two domains. The work in [16] attempted to find a projection matrix to transfer source and target domain data into a common feature subspace. Then, joint low-rank constraint and sparse representation were applied to the reconstructed matrix to preserve local and global data structures. In [17], an unsupervised steganalysis method based on subspace learning was proposed, the global and local structures of data were maintained by the low-rank and sparse constraints of the reconstruction coefficient matrix to obtain a new feature representation. In this way, the feature distributions of training data and testing data are close to each other. For JPEG recompression, multiple classifiers were constructed to detect the recompression Markov characteristics of the testing images, and the steganalysis features were then transferred to a new feature subspace [18]. A feature transfer algorithm based on contribution was designed in [19], which attempted to transfer training set features by evaluating the contribution of sample features and dimension features.

For the second kind of method, a domain adaptation classifier was constructed, which directly integrated domain adaptation principles as regularization terms. For instance, by adding conditional distribution into the Laplacian regularization, ARTL [26] was improved by [20]. Joint distribution adaptation and geometric structure were also integrated into the domain adaptation classifier. In the spatial domain, the joint distribution of input image and prediction label were considered simultaneously in [21]. In their experiment, when a model trained on a dataset with low texture complexity detects a test set with high texture complexity, the accuracy of steganalysis will decrease significantly. J-NET proposed in [21] alleviates this decline to a certain extent.

In addition to the aforementioned methods, a novel approach that can effectively measure texture complexity was proposed by [22]. The authors improved the accuracy of steganalysis by finding the block closest to the Most Effective Range (MER) in an image to represent the whole image. Concerning payload mismatch, a general idea [23,24] was to first train the model at a high payload and further fine-tune the model with new data at low payload asymptotically. Finally, a steganalysis model that can detect steganographic methods at a low embedding rate is obtained. Although these methods can reduce the impact of CSM to a certain extent, there is still room for improvement.

First, most of the existing steganalysis schemes are based on deep learning. However, to the best of our knowledge, the cover source mismatch problems solved by [16-22] are all for hand-crafted features, such as Pevny Method (PEV) [27], Rich Model [28], CC-PEV [29], and DCTR [30]. Few works [21][31] have been conducted on cover source mismatch in deep learning-based steganalysis. Another observation is that the existing solutions mostly follow a strategy of global alignment, which cannot effectively enlarge the distance between different classes. In our case, however, a subdomain aligning strategy is adopted to describe a clearer margin.

3 Motivation

The cover source mismatch in steganalysis is similar to the domain adaptation problem. Domain adaptation focus on solving the problem of inconsistent feature distribution, where feature space and category space are consistent [32]. In domain adaptation, the source domain represents a domain with rich supervisory information and the target domain is the domain of testing samples where there are no labels or only a few labels. For a target task, we collect a batch of unlabeled data from the target domain and labeled data from the source domain. How to use these data to get a well-behaved model for the target task is the problem to be solved.

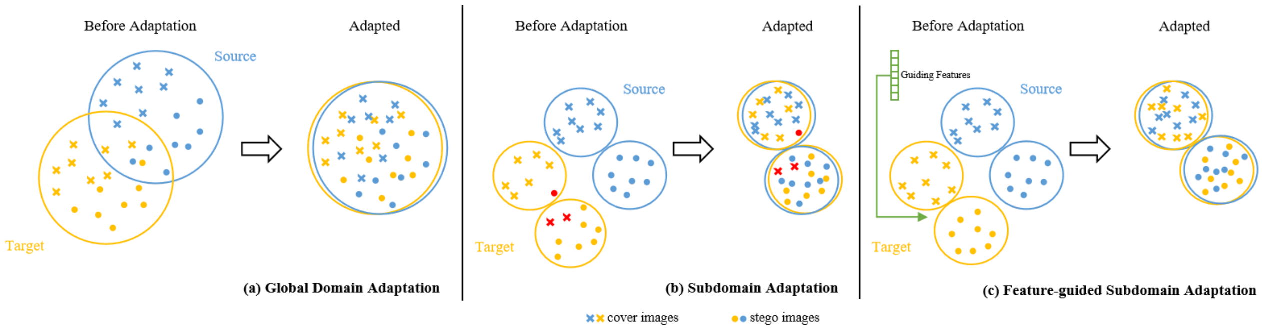


Fig. 2 The comparison of several relevant domain adaptation processes

The most common metric in domain adaptation is the Maximum Mean Discrepancy (MMD) [33], which maps two primitive variables to the Reproducing Kernel Hilbert Space (RKHS) and then measures the distance between the two distributions in RKHS. Based on MMD, many domain adaptation methods have been proposed, such as single kernel method deep domain confusion (DDC) [34], multi-kernel method deep adaptation network (DAN) [35], joint adaptation networks (JAN) [36], and so on. Recently, more works have focused on subdomain adaptation [37-39]. However, most of them are adversarial methods with multiple loss functions and slow convergence. A non-adversarial deep subdomain adaptation network (DSAN) [40] was proposed for image classification to align the distributions of correlated subdomains. Compared with the adversarial method, the method in [40] was more simple and more efficient, which also achieved comparable results with the adversarial method [37]. Due to the advantages and feasibility of DSAN, we introduced and optimized it into the image steganalysis field to solve the dataset mismatch problem.

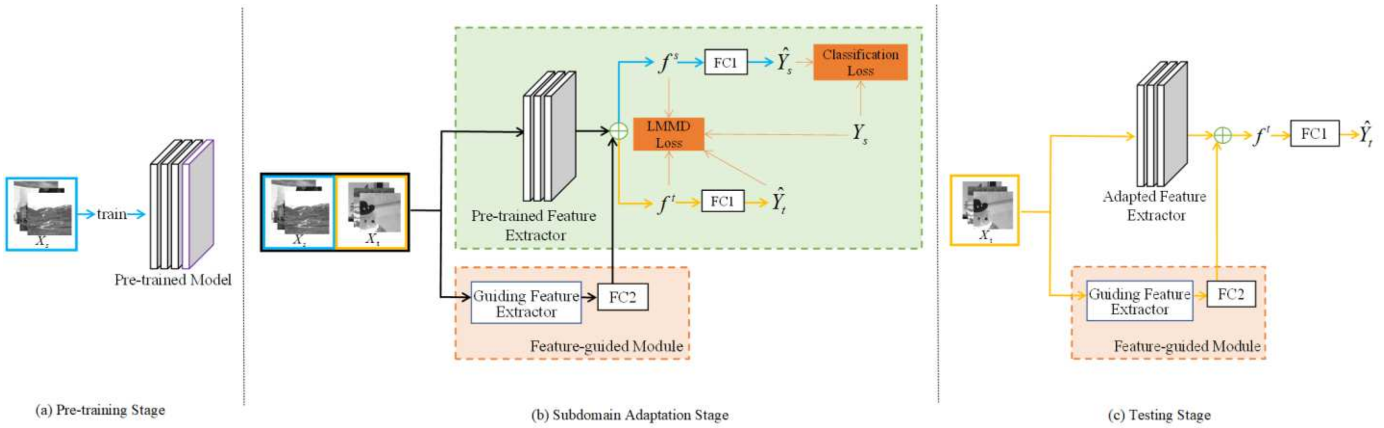
DSAN is an image classification method that is quite different from image steganalysis, where image steganalysis focuses on the subtle traces instead of content sematic. When the model does not perform well in the target domain, such as at low embedding rate or under severe mismatch, it is not reliable to directly apply DSAN method. A guiding feature module is designed to deal with this situation. The advantages of our proposed steganalysis framework are illustrated intuitively in Fig. 2. More specifically, (a) The global methods make the classifier miss a lot of fine-grained information for each class. As a result, the classifier cannot adequately describe the boundary between two different classes. (b) By dividing domains into subdomains, the distance between classes is increased and the distance within classes is decreased. Note that the division of subdomains largely depends on the prediction given by the model in target domain. However, mismatch has already existed, i.e. the predictions given by the model in target domain are probably wrong which will lead to a bad subdomain division (red points). (c) To correct the division, we design guiding feature to guide the steganalysis network to obtain more robust pseudo labels and further urge the model to transfer in the right direction.

4 Proposed framework

4.1 Notation

Before introducing the method, we first express the dataset mismatch problem in image steganalysis as follows. Let $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ represents the source domain, which contains n_s samples with labels. y_i is a label vector with two dimensions, where its first component corresponds to the probability to be a cover image, and its second dimension corresponds to the probability to be a stego image. When y_i is related to samples on the source domain, y_i =[1,0] or [0,1]. On the other hand, when y_i is related to samples on the target domain, y_i owns a soft label with the sum of all components equal to 1. $D_t = \{x_j^t\}_{j=1}^{n_t}$represents the target domain, which contains n_t samples without labels. The marginal probability distributions of D_s and D_t are p and q, where $p \neq q$. In subdomain adaptation, the source domain D_s and the target domain D_t are divided into $D_s^{(c)}$ and $D_t^{(c)}$ according to class c, respectively. The main purpose of this paper is to reduce the change of distributions of related subdomains and make full use of the training set with labeled data to correctly predict the test set without labeled data.

4.2 Architecture



(a) Pre-training Stage

(b) Subdomain Adaptation Stage

(c) Testing Stage

Fig. 3 The overall procedure of the proposed steganalysis framework

As shown in Fig. 3, the overall steganalysis procedure consists of three stages, including the pre-training stage, subdomain adaptation stage, and testing stage.

1. **Pre-training Stage**

   In the pre-training stage, the model is supervised trained on the dataset of the source domain. After training, a pre-trained model *M* is obtained. The performance of *M* on other datasets is usually not as good as that in the training set. *M* consists of a pre-trained feature extractor *P* and pre-trained classification layers. The features derived from higher levels of the network must depend to a large extent on specific datasets and tasks, which cannot be safely transferred to new tasks [35]. So the pre-trained classification layers are abandoned. Such as the SRNet steganalysis model, the last Fully Connected (FC) layer is abandoned.

2. **Subdomain Adaptation Stage**

   The steganalysis framework we proposed works at this stage. Both source domain data and target domain data participate in the adaptation training. The data flows of source domain and target domain are represented in blue and yellow lines, respectively, as shown in Fig. 3 (b). The feature f_P is extracted through the pre-trained feature extractor *P*. On the other hand, the hand-crafted features f_G is extracted through the guiding-feature extractor *G*. f_G is selected and dimensionally reduced to feature $f_G{}'$ by *FC2*. f_P and $f_G{}'$ are concatenated to obtain f. f is used to provide the final prediction. The weight of each sample belonging to class *c* is calculated for subdomain division using Eq. (4). X_s use true labels Y_s and X_t use the prediction result of *FC1*. The discrepancy between different subdomains belonging to the same class is narrowed by minimizing LMMD [40] loss. The details are shown in Pseudocode 1.

3. **Testing stage**

   After training, the suspicious input is sent to the adapted feature extractor and the feature-guided module simultaneously. The concatenated feature is generated and used to obtain the final classification result.

Pseudocode 1 Feature-guided deep subdomain adaptation

```
Input: Pre-trained feature extractor $P$, guiding feature extractors $G$, source domain data $X_s$, target domain data $X_t$, source domain labels $Y_s$, $FC1$ for classification, $FC2$ for dimension reduction.
Output: Adapted steganalysis model
for each epoch do
    for each batch do
        //Extract features using $P$.
        $f_P^s = P(X_s)$
        $f_P^t = P(X_t)$
        //Extract features using $G$.
        $f_G^s = G(X_s)$
        $f_G^t = G(X_t)$
        //Select and dimensionally reduce the guiding feature using a FC layer $FC2$.
        $f_G^{s\prime} = FC2(f_G^s)$
        $f_G^{t\prime} = FC2(f_G^t)$
        //Concatenate two features.
        $f^s$ = concatenate ($f_P^s$, $f_G^{s\prime}$)
        $f^t$ = concatenate ($f_P^t$, $f_G^{t\prime}$)
        //Feed the concatenated features $f^s$ and $f^t$ into the classification layers $FC1$ to obtain the prediction result.
        $\hat{Y}_s = FC1(f^s)$
        $\hat{Y}_t = FC1(f^t)$
        // Calculate the classification loss of source domain.
        classification_loss= BCE ($\widehat{Y_s}$, $Y_s$)
        // Calculate the LMMD loss between two domains.
        lmmd_loss = LMMD ($f^s$, $f^t$, $Y_s$, $\widehat{Y_t}$)
        // Calculate the total loss.
        loss = classification_loss + $\lambda * $ lmmd_loss
        Backpropagation
        Update the parameters of P, FC1 and FC2.
    end for
end for
```

4.3 Local alignment and LMMD

In this paper, we adopt a local alignment strategy instead of the global alignment strategy to better reduce the distance in feature distributions between domains. The global alignment strategy tends to make the global distribution consistent, but cannot give a clear decision boundary after adaption. However, by applying a local alignment strategy, the distance between data of the same class in different domains is narrowed. Meanwhile, the distance between different classes is effectively enlarged, which is conducive to the improvement of classification accuracy.

In subdomain adaptation, we divide the subdomains according to the labels of the data in the source domain, while in the target domain, the data has no labels. Therefore, the prediction of the model is used as the basis for the division of the target domain. Also, to align the distributions of related subdomains, we need a metric that can estimate the distribution differences between subdomains. In this paper, we applied LMMD [40], which is a measure based on MMD. Different from MMD, LMMD can measure differences in local distributions. It is defined as

$$d_H(p,q) \triangleq E_c \left\| E_{p^{(c)}}[\varphi(x^s)] - E_{q^{(c)}}[\varphi(x^t)] \right\|_H^2 \qquad (1)$$

where H is the RKHS endowed with a characteristic kernel k. $E_c[\cdot]$ is the mathematical expectation of class c. $p^{(c)}$ and $q^{(c)}$ are the distributions of $D_s^{(c)}$ and $D_t^{(c)}$. $\varphi(\cdot)$ is the feature map which project the original samples from D_s and D_t to RKHS.

4.4 Feature-guided module

In domain adaptation, the data of the target domain has no labels, so we need to use the pseudo labels predicted by the model to divide the subdomains. However, models tend not to perform as well in the target domain as they do in the source domain since the feature distributions are different. Although soft labels (i.e. the probability distribution predicted by the model) were used to better utilize the output from the model, if the model itself performs poorly in the target domain due to large differences in sample distribution that leads to many wrong results, it is very likely that high accuracy will not be obtained even after alignment. This leads us to search for a way that generates more robust pseudo labels for the target domain to better help models perform domain adaptation.

In order to enhance the quality of pseudo labels, guiding feature is designed. In this paper, SRM [28] and maxSRMd2 [41] are chosen to generate guiding features due to their good performance in steganalysis. SRM uses a variety of linear and nonlinear high-pass filters and calculates the four-dimensional co-occurrence matrix of the obtained features. Finally, a 34,671-dimensional feature vector is obtained. The maxSRMd2 is a variant of the SRM that makes use of the selection channel.

In the experimental results of [42], feature selection has been proved to be a good tool to deal with the CSM problem. It can remove irrelevant features which could disrupt the classification process. In our work, an FC layer is used to select features from the hand-crafted feature, such as SRM and maxSRMd2, and also reduce their dimension. The selected features are concatenated with the features extracted by the pre-trained feature extractor. Then, the concatenated features are sent to the classification layer for label prediction. Relevant experiments are performed to select the most appropriate dimension of guiding features. With the guidance of prior knowledge, we can avoid the misdirection of wrong labels to the model, and make the subdomain division more accurate. Hence, better aligns the subdomains and reduces the difference of feature distributions among the subdomains.

4.5 Loss function

In general, the loss function of a domain adaptation classifier is composed of two terms, classification loss and domain adaptation loss.

$$\min_f \frac{1}{n_s}\sum_{i=1}^{n_s} J(f(x_i^s), y_i^s) + \lambda\hat{d}(p,q) \qquad (2)$$

where the former is the cross-entropy loss function between the classifier $f(x_i^s)$ and the labels y_i^s. And $\lambda\hat{d}(p,q)$ is the domain adaptation loss which is often defined differently in different works. $\lambda > 0$ is the trade-off parameter of the domain adaptation loss and the classification loss. By minimizing Eq. (2) during the training phase of the network parameters, the difference between feature distributions is reduced.

Assuming that each sample is classified according to weight w^c. Then, the unbiased estimator of Eq. (1) can be written as

$$\hat{d}_H(p,q) = \frac{1}{C}\sum_{c=1}^{C}\left\|\sum_{x_i^s\in D_s} w_i^{sc}\varphi(x_i^s) - \sum_{x_j^t\in D_t} w_j^{tc}\varphi\left(x_j^t\right)\right\|_H^2 \tag{3}$$

where w_i^{sc} and w_j^{tc} denote the weights of x_i^s and x_j^t belonging to class *c*. The weighted sample x_i belongings to class c is

$$w_i^c = \frac{y_{ic}}{\sum_{(x_j,y_j)\in D} y_{jc}} \tag{4}$$

where y_{ic} is the c_{th} element of the vector y_i. For the source domain, w_i^c is a discrete value related to the true labels. For the target domain without labels, we use soft labels that predicted by the concatenated feature instead of hard labels. This is because soft labels contain more information than hard labels. The data that came from D_s and D_t will be activated as $\{z_i^{sl}\}_{i=1}^{n_s}$ and $\{z_j^{tl}\}_{j=1}^{n_t}$ after layer *l*. Kernel trick is used to map indivisible data from low dimensional space to high dimensional space. The characteristic kernel k is associated with the feature map $\varphi(\cdot)$. The application of the kernel $k(x^s, x^t)$ is equal to the inner product of vectors $\varphi(x^s)$ and $\varphi(x^t)$. Eq. (3) can be calculated in the following way.

$$\begin{aligned}\hat{d}_l(p,q) = \frac{1}{C}\sum_{c=1}^{C}\Bigg[&\sum_{i=1}^{n_s}\sum_{j=1}^{n_s} w_i^{sc}w_j^{sc}k(z_i^{sl}, z_j^{sl}) \\ &+\sum_{i=1}^{n_t}\sum_{j=1}^{n_t} w_i^{tc}w_j^{tc}k(z_i^{tl}, z_j^{tl}) \\ &-2\sum_{i=1}^{n_s}\sum_{j=1}^{n_t} w_i^{sc}w_j^{tc}k(z_i^{sl}, z_j^{tl})\Bigg]\end{aligned} \tag{5}$$

Eq. (5) can be used as domain adaptation loss directly. The final loss function of the final classifier in layer *l* can be obtained by substituting Eq. (5) into Eq. (2)

$$\min_f \frac{1}{n_s}\sum_{i=1}^{n_s} J(f(x_i^s), y_i^s) + \lambda\sum_{l\in L}\hat{d}(p,q) \tag{6}$$

Through Eq. (6), we can reduce the difference in the distributions of related subdomains of activation layer. The labels of the target domain become more accurate during the iteration process.

5 Experimental evaluation

5.1 Dataset

To evaluate the capability of models in dataset mismatch scenarios, we select four different widely-used image datasets. Here, we first give a brief introduction to these datasets.

BOSSbase 1.01 [43] consists of 10,000 uncompressed grayscale images. Image size is 512 x 512 and images come from seven different digital cameras. It is the most commonly used dataset in steganography and steganalysis.

UCID.v2 [44] is a dataset containing 1338 uncompressed TIFF images on a variety of topics, including indoor and outdoor natural scenes and man-made objects. The image size is 512×384 or 384×512, and all images were taken with a Dimage 5 color digital camera.

DIV 2K [45] is a diverse dataset containing 1000 images in total. All 1000 images are 2K resolution, meaning they have at least 2K pixels on one axis. These images are all hand-pulled color RGB images from dozens of websites. All images are processed using the same tools and are formatted as PNG.

MIR FLICKR25K [46] contains 25000 JPEG images, these images are very diverse and close to the authentic images in real life.

In the pre-training stage, images from BOSSbase 1.01 were resized to 256×256 and used to train the pre-trained steganalysis models. In the subdomain adaptation stage, 500 randomly selected image pairs from the BOSSbase 1.01 were used as source domain data. Meanwhile, 500 pairs of images were selected randomly from the other three datasets as the target domain data, respectively. Due to the difference in the number and size of images, we processed the other three datasets differently. For images from UCID, they were cropped in the four corners with size 256×256. Then they are converted to grayscale images and saved in PGM format. For images from DIV2K, are first resized to 512×512 and then cropped in four corners. For those from Flickr 25K, they are first converted into PGM grayscale images, resized to 512×512, and then centered to 256×256. It is noted that most of the original image size is less than 512×512. Therefore, we believe that the quality of the processed MIR FLICKR25K image will be lower after down-sampling, (i.e. the texture complexity will be lower).

Our experimental dataset is finally composed of 500 pairs of BOSSbase images as the source domain, and 500 pairs of images from Flickr25K, UCID, and DIV 2K, respectively, as the target domain. It is worth mentioning that the data of the target domain is unlabeled, that is to say, the classifier cannot update the parameters of the network by using the labels of the target domain.

5.2 Experiment setup

In our experiment, three steganographic methods, S-UNIWARD, WOW, and HUGO, are used to generate the stego images. And the deep learning-based model we used is SRNet. In the works of [4-7] and [13], the authors trained the model on the BOSSBase 1.01 dataset, and we will continue with this practice. The reason for choosing SRNet is that it is an end-to-end deep learning network that does not introduce too many hand-designed elements. Therefore, the SRNet model trained on BOSSBase 1.01 dataset was chosen as the representative of steganalysis in the "laboratory environment". We conducted experiments at both 0.4bpp (bits per pixel) 0.2bpp and 0.1bpp to verify the effectiveness of our proposed steganalysis framework.

Note that we did not use data augmentation in the pre-training stage, and the rest of the settings followed the paper [13]. According to [13], the 0.2bpp model was obtained by fine-tuning the 0.4bpp model for 100 epochs. Similarly, the 0.1bpp model was obtained by fine-tuning the 0.2bpp model. Finally, we obtained three trained SRNet models at the embedding rates of 0.1bpp, 0.2bpp and 0.4bpp, respectively.

For all tasks, we use the small batch SGD with momentum 0.9 and the learning rate annealing strategy in [40]. During the subdomain adaptation stage, we set the batch size to 16. The learning rate is initialized to 0.01 and updated by $\eta_\theta = \eta_0/(1+\alpha\theta)^\beta$, where θ is a linear change in the range of 0 to 1 during training, $\eta_0 = 0.01$, $\alpha = 10$, and $\beta = 0.75$. With this strategy, the learning rate will be reduced to 0.0017 after the whole 200 epochs. Instead of fixing the adaptation factor λ, we gradually change it from 0 to 1 over an incremental schedule, $\lambda_\theta = 2/exp(-\gamma\theta) - 1$. This is done in order to suppress unstable pseudo labels in the model output at the initial stage of training. $\gamma = 10$ is fixed throughout the experiment which is also compliant to [40]. All the experiments in this paper, including SRNet training, are implemented in PyTorch. The Gaussian kernel is used in our experiment, other kernels that is able to do spatial mapping can also be used.

5.3 Experimental results

The experiments mainly include six parts. In the first part, we conduct a comparative analysis on the selection of SRM feature dimension. In the second part, we compare with other state-of-the-art methods on three different datasets at 0.1bpp, 0.2bpp and 0.4bpp payload. In the third part, we modify the network structure to verify whether our framework is still effective after the network structural changes. In the fourth part, an ablation experiment is provided to emphasize the importance of guiding feature. In the fifth part, we conduct parameter sensitivity experiments. In the last part, another guiding feature is applied to verify the universality of the feature-guided module. The detection performance is measured with the average total accuracy $P_A = \frac{1}{2} ave(p_c + p_s)$, which is also used in [20]. p_c is the classification accuracy rate of cover images, and p_s is the detection accuracy rate of stego images. We repeat all the domain adaptation experiments (J-Net, DSAN, ours) three times and take their average value as the final result.

5.3.1 Selection of the SRM features dimension

As we mentioned in Section 4.4, the extracted hand-crafted features should be selected and dimensionally reduced to form guiding feature by a FC layer. The accuracy when reducing the SRM feature to different dimensions is tested, and the results are shown in Fig. 4. The experiment is completed by using UCID.v2 images as the target domain because it is on this

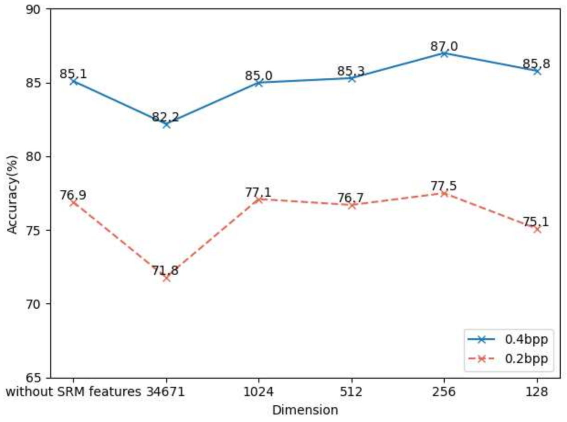


Fig. 4 Detection accuracy of our framework when selecting different SRM feature dimensions (%)

dataset that the accuracy drops the most. And the steganographic method used is S-UNIWARD. It can be observed that there is a sharp drop when fusing SRM features directly which proves the importance of feature selection. The performances of other feature dimensions are similar, and the accuracy of fusing 256-dimensional features has the highest increase which is about 2% higher than the one without guiding features at 0.4bpp and 0.5% at 0.2bpp. Introducing guiding features improves accuracy more significantly at 0.4bpp than 0.2bpp. In the end, we chose to retain 256-dimensional guiding features.

5.3.2 Comparison with other state-of-the-art methods

Through experiments, we found that when the model trained on BOSSbase detects the data from other datasets, the accuracy of the model will decrease by about 3%-7%. Exceptions occur when detecting the S-UNIWARD and HUGO pairs from Flickr 25K. In these cases, the accuracy of the model actually increased. This is probably because of the processing we did with Flickr 25K data which resulted in the lower texture complexity of the images (see Section 5.1). Therefore, the model trained on BOSSbase with high texture complexity without format conversion, down-sampling, and other processing can still effectively detect stego images with low texture complexity under the condition of dataset mismatch.

Table 1, Table 2 and Table 3 present the experimental results in the case of dataset mismatch at the payload of 0.4bpp, 0.2bpp and 0.1bpp, respectively. The SRNet column represents the accuracy of the model in the case of dataset mismatch. It can be seen that our framework can significantly reduce the impact of dataset mismatch while improving the average accuracy by 6.0% at 0.4bpp, 8.5% at 0.2bpp and 8.0%at 0.1bpp. The accuracy of our proposed steganalysis framework is higher than that of J-Net [21] at various embedding rates.

Table 1 Detection accuracy under dataset mismatch at 0.4bpp (%)

Dataset	Steganography	SRNet	J-Net	Ours
UCIDv.2	SUNI	81.6	83.0	**87.0**
	WOW	81.0	82.3	**87.0**
	HUGO	78.8	81.7	**85.4**
DIV2K	SUNI	83.5	84.5	**87.6**
	WOW	80.5	81.4	**88.0**
	HUGO	80.1	81.7	**88.1**
Flickr 25K	SUNI	87.4	87.9	**92.3**
	WOW	84.3	86.5	**90.5**
	HUGO	88.4	91.0	**92.3**
Average		82.8	84.4	**88.2**

Table 2 Detection accuracy under dataset mismatch at 0.2bpp (%)

Dataset	Steganography	SRNet	J-Net	Ours
UCIDv.2	SUNI	70.3	74.8	**77.5**
	WOW	68.2	67.8	**79.1**
	HUGO	70.6	70.7	**75.8**
DIV2K	SUNI	71.6	75.9	**79.5**
	WOW	69.9	70.4	**81.4**
	HUGO	70.2	70.9	**79.9**
Flickr 25K	SUNI	76.8	79.6	**84.4**
	WOW	71.0	74.4	**80.3**
	HUGO	78.3	81.0	**85.6**
Average		71.9	73.9	**80.4**

Table 3 Detection accuracy under dataset mismatch at 0.1bpp (%)

Dataset	Steganography	SRNet	J-Net	Ours
UCIDv.2	SUNI	61.0	60.9	**69.7**
	WOW	60.3	60.2	**66.2**
	HUGO	61.0	61.2	**63.8**
DIV2K	SUNI	63.2	63.5	**74.7**
	WOW	59.9	60.5	**68.1**
	HUGO	61.9	62.9	**68.3**
Flickr 25K	SUNI	65.0	67.0	**76.2**
	WOW	61.7	63.6	**69.3**
	HUGO	67.5	68.0	**77.0**
Average		62.4	63.1	**70.4**

5.3.3 Effect to transfer due to changes in structure

To verify whether our framework is still effective after the network structure changes, we modified the network structure by cutting layer 4 to layer 7 of the SRNet model,

Table 4 Detection accuracy under dataset mismatch when the model structure changes at 0.4bpp (%)

Dataset	Steganography	Clipped SRNet	Ours
UCIDv.2	SUNI	71.5	85.4
	WOW	74.0	85.6
	HUGO	76.4	83.0
DIV2K	SUNI	81.2	86.4
	WOW	77.1	86.8
	HUGO	77.6	84.0
Flickr 25K	SUNI	86.6	93.4
	WOW	85.0	92.2
	HUGO	82.2	91.2
Average		79.1	87.6

Table 5 Detection accuracy under dataset mismatch when the model structure changes at 0.2bpp (%)

Dataset	Steganography	Clipped SRNet	Ours
UCIDv.2	SUNI	67.3	74.5
	WOW	65.8	79.5
	HUGO	63.8	74.1
DIV2K	SUNI	69.1	78.1
	WOW	66.7	81.2
	HUGO	62.9	70.9
Flickr 25K	SUNI	71.8	84.8
	WOW	69.4	82.3
	HUGO	65.2	80.6
Average		66.9	78.4

Table 6 Detection accuracy of ablation experiments (%)

Payload		0.4bpp		0.2bpp		0.1bpp	
Dataset	Steganography	DSAN	FG-SAS	DSAN	FG-SAS	DSAN	FG-SAS
UCID v.2	SUNI	85.1	**87.0**	76.9	**77.5**	60.4	**69.7**
	WOW	86.0	**87.0**	77.3	**79.1**	58.5	**66.2**
	HUGO	84.6	**85.4**	**76.4**	75.8	59.2	**63.8**
DIV 2K	SUNI	86.7	**87.6**	**80.7**	79.5	58.6	**74.7**
	WOW	85.6	**88.0**	79.5	**81.4**	53.9	**68.1**
	HUGO	84.9	**88.1**	77.0	**79.9**	58.3	**68.3**
Flickr 25K	SUNI	**92.7**	92.3	**84.9**	84.4	72.8	**76.2**
	WOW	89.9	**90.5**	78.5	**80.3**	65.0	**69.3**
	HUGO	**93.8**	92.3	85.4	**85.6**	74.5	**77.0**
Average		87.7	**88.2**	79.6	**80.4**	62.4	**70.4**

and re-train the clipped model to obtain new pre-trained models. We found that the accuracy of the clipped model is close to the original model when there is no mismatch. However, the accuracy of the clipped one drops more when mismatch happened compared with the original models. We transferred the new models as we did in Section 5.3.2 and the results are shown in Table 4 and Table 5. It can be observed that the average accuracy of the original model can be increased by about 8% in our framework at 0.4bpp and 11% at 0.2bpp. Our framework is still effective when the structure of the model changes.

5.3.4 Ablation experiments

To verify the importance of our feature guiding strategy, ablation experiments are provided in this part. The results are shown in Table 6, DSAN denotes the method only applying subdomain adaptation and FG-SAS denotes the proposed feature-guided subdomain adaptation steganalysis framework. FG-SAS achieves higher accuracy in most cases than DSAN. Note that when the pre-trained model performs poorly in the target domain, such as 0.1bpp, it is not reliable to divide the subdomains correctly according to the prediction result, which may make the model transfer in a negative direction. However, by selecting and dimensionally reducing the hand-crafted feature, guiding features can help with the stability of adapting, and higher accuracy can be achieved, which verifies the significance of feature guiding strategy for steganalysis with cover source mismatch.

5.3.5 Sensitivity analysis of parameter

In this part, we conduct sensitivity experiments on the trade-off parameter λ in Eq. (6) between domain adaptation loss and classification loss. When different trade-off parameters were selected, the detection accuracy of the model is shown in Fig. 5. The experiment is completed by using UCID.v2 images as the target domain and the S-UNIWARD steganographic method.

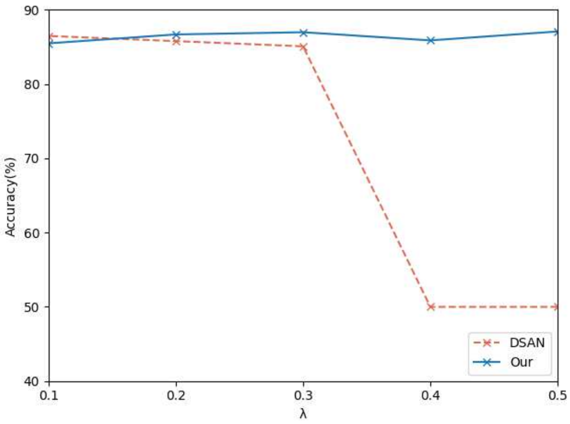


Fig. 5 Detection accuracy under different trade-off parameter values.

It can be seen that the improved model is not very sensitive to the change of trade-off parameters, while the DSAN that has not integrated guiding features is more sensitive to the setting of the trade-off parameter λ. When λ is 0.4 and 0.5, the performance of DSAN is equivalent to a random guess, which means that the model did not work at all. This did not happen until the model has been transferring for a while. It is probably because the weight of adaptation loss is higher and the predictions of models in the target domain are given a higher level of confidence. As a result, the model tends to transfer in the wrong direction. However, the model guided with the select set of SRM features can correct this error to some extent. Therefore, our steganalysis framework is more stable than DSAN.

5.3.6 Discussion on guiding features

In order to demonstrate the universality of the feature-guided module, we also complete our experiment by using maxSRMd2 as the guiding feature extractor. The experimental results are shown in Table 7 and Table 8. The steganographic method we used in this part is S-UNIWARD. The model transferred with maxSRMd2 will be denoted as “Ours_max” to distinguish it from the one with SRM (denoted as “Ours”). It can be seen that the results are similar while transferring with the guidance of the select set of SRM or maxSRMd2. Our framework is still effective for other well-behaved guiding features.

Table 7 Detection accuracy using different guiding features at 0.4bpp (%)

Dataset	SRNet	Ours	Ours_max
UCIDv.2	81.6	87.0	86.3
DIV2K	83.5	87.6	88.6
Flickr 25K	87.4	92.3	91.4

Table 8 Detection accuracy using different guiding features at 0.2bpp (%)

Dataset	SRNet	Ours	Ours_max
UCIDv.2	70.3	77.5	75.9
DIV2K	71.6	79.5	77.8
Flickr 25K	76.8	84.4	86.3

6 Conclusion and future work

In this paper, a feature-guided subdomain adaptation framework is proposed to overcome the challenge of dataset mismatch in deep learning-based steganalysis. In practical communication application scenarios, we are inevitably faced with a variety of cover source mismatch problems, which makes the deployment of steganalysis tools very difficult. Because once there is a difference in the feature distribution between the training set and the test set, the performance of the steganalysis will be effected. Different from the previous steganalysis methods of subspace transfer learning and global alignment, we adopted the subdomain alignment method to make full use of fine-grained information, better mine the relationship between the source domain and target domain data, and reduce the differences between domains. To divide the subdomains more accurately, guiding features were responsible for resisting the influence of the wrong prediction on domain adaptation. Our experimental results on multiple datasets, against a variety of steganographic methods and different payloads, showed that our framework can effectively help the steganalysis models adapt to the new data distribution and make better classification. Even though the structure of the network has changed, our approach is still effective. The purpose of this paper is to bring steganalysis from the laboratory to the real-world communication application, so as to reduce the difficulties brought by the real scene to the deep learning-based steganalysis networks.

Our proposed framework is currently suitable for dataset mismatch in spatial steganalysis. Dataset mismatch is a scenario of cover source mismatch and we expect that our framework can be used for steganographic methods mismatch and payload mismatch. Moreover, the performance of our framework in the JPEG domain is also worth studying due to the widespread use of JPEG images.

Acknowledgments

This work was supported by National Natural Science Foundation of China (61972269, 61902263), and Sichuan Science and Technology Program (22ZDYF3635).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Holub, V., Fridrich, J., & Denemark, T. (2014). Universal distortion function for steganography in an arbitrary domain. *EURASIP Journal on Information Security*, 2014(1), 1-13. https://doi.org/10.1186/1687-417X-2014-1
2. Holub, V., & Fridrich, J. (2012). Designing steganographic distortion using directional filters. *2012 IEEE International Workshop on Information Forensics and Security* (pp. 234-239). https://doi.org/10.1109/WIFS.2012.6412655
3. Pevný, T., Filler, T., & Bas, P. (2010). Using high-dimensional image models to perform highly undetectable steganography. *International Workshop on Information Hiding* (pp. 161-177).
4. Xu, G., Wu, H. Z., & Shi, Y. Q. (2016). Structural design of convolutional neural networks for steganalysis. *IEEE Signal Processing Letters*, 23(5), 708-712. https://doi.org/10.1109/LSP.2016.2548421
5. Ye, J., Ni, J., & Yi, Y. (2017). Deep learning hierarchical representations for image steganalysis. *IEEE Transactions on Information Forensics and Security*, 12(11), 2545-2557. https://doi.org/10.1109/TIFS.2017.2710946
6. Yedroudj, M., Comby, F., & Chaumont, M. (2018). Yedroudj-

net: An efficient CNN for spatial steganalysis. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 2092-2096). https://doi.org/10.1109/ICASSP.2018.8461438

7. Zhang, R., Zhu, F., Liu, J., & Liu, G. (2020). Depth-wise separable convolutions and multi-level pooling for an efficient spatial CNN-based steganalysis. *IEEE Transactions on Information Forensics and Security*, 15, 1138-1150. https://doi.org/10.1109/TIFS.2019.2936913

8. Zou, D., Shi, Y. Q., Su, W., & Xuan, G. (2006). Steganalysis based on Markov model of thresholded prediction-error image. *2006 IEEE International Conference on Multimedia and Expo* (pp. 1365-1368). https://doi.org/10.1109/ICME.2006.262792

9. Pevny, T., Bas, P., & Fridrich, J. (2010). Steganalysis by subtractive pixel adjacency matrix. *IEEE Transactions on Information Forensics and Security*, 5(2), 215-224. https://doi.org/10.1109/TIFS.2010.2045842

10. Holub, V., Fridrich, J., & Denemark, T. (2013). Random projections of residuals as an alternative to co-occurrences in steganalysis. *Media Watermarking, Security, and Forensics 2013* (Vol. 8665, p. 86650L). International Society for Optics and Photonics.

11. Luo, X., Liu, F., Lian, S., Yang, C., & Gritzalis, S. (2011). On the typical statistic features for image blind steganalysis. *IEEE Journal on Selected Areas in Communications*, 29(7), 1404-1422. https://doi.org/10.1109/JSAC.2011.110807

12. Wang, Y., Liu, J., & Zhang, W. (2009). Blind JPEG steganalysis based on correlations of DCT coefficients in multi-directions and calibrations. *2009 International Conference on Multimedia Information Networking and Security* (Vol. 1, pp. 495-499). https://doi.org/10.1109/MINES.2009.135

13. Boroumand, M., Chen, M., & Fridrich, J. (2019). Deep residual network for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 14(5), 1181-1193. https://doi.org/10.1109/TIFS.2018.2871749

14. Kodovský, J., Sedighi, V., & Fridrich, J. (2014). Study of cover source mismatch in steganalysis and ways to mitigate its impact. *Media Watermarking, Security, and Forensics 2014* (Vol. 9028, p. 90280J). International Society for Optics and Photonics. https://doi.org/10.1117/12.2039693

15. Giboulot, Q., Cogranne, R., Borghys, D., & Bas, P. (2020). Effects and solutions of cover-source mismatch in image steganalysis. *Signal Processing: Image Communication*, 86, 115888. https://doi.org/10.1016/j.image.2020.115888

16. Jia, J., Zhai, L., Ren, W., Wang, L., Ren, Y., & Zhang, L. (2020). Transferable heterogeneous feature subspace learning for JPEG mismatched steganalysis. *Pattern Recognition*, 100, 107105. https://doi.org/10.1016/j.patcog.2019.107105

17. Xue, Y., Yang, L., Wen, J., Niu, S., & Zhong, P. (2019). A subspace learning-based method for JPEG mismatched steganalysis. *Multimedia Tools and Applications*, 78(7), 8151-8166. https://doi.org/10.1007/s11042-018-6719-5

18. Yang, Y., Kong, X., & Feng, C. (2018). Double-compressed JPEG images steganalysis with transferring feature. *Multimedia Tools and Applications*, 77(14), 17993-18005. https://doi.org/10.1007/s11042-018-5734-x

19. Feng, C., Kong, X., Li, M., Yang, Y., & Guo, Y. (2017). Contribution-based feature transfer for JPEG mismatched steganalysis. *2017 IEEE International Conference on Image Processing (ICIP)* (pp. 500-504). https://doi.org/10.1109/ICIP.2017.8296331

20. Yang, Y., Kong, X., Wang, B., Ren, K., & Guo, Y. (2019). Steganalysis on Internet images via domain adaptive classifier. *Neurocomputing*, 351, 205-216. https://doi.org/10.1016/j.neucom.2019.04.025

21. Zhang, X., Kong, X., Wang, P., Wang, B. (2019). Cover-source Mismatch in Deep Spatial Steganalysis. *Proceedings of 18th Workshop on Digital Forensics and Watermarking*, pp. 71-83, 2019. http://dx.doi.org/10.1007/978-3-030-43575-2_6

22. Hu, D., Ma, Z., Fan, Y., Zheng, S., Ye, D., & Wang, L. (2019). Study on the interaction between the cover source mismatch and texture complexity in steganalysis. *Multimedia Tools and Applications*, 78(6), 7643-7666. https://doi.org/10.1007/s11042-018-6497-0

23. Ozcan, S., & Mustacoglu, A. F. (2018). Transfer learning effects on image steganalysis with pre-trained deep residual neural network model. *2018 IEEE International Conference on Big Data (Big Data)* (pp. 2280-2287). https://doi.org/10.1109/BigData.2018.8622437

24. El Beji, R., Saidi, M., Hermassi, H., & Rhouma, R. (2018). An Improved CNN Steganalysis Architecture Based on "Catalyst Kernels" and Transfer Learning. *International Conference on Digital Economy* (pp. 119-128). https://doi.org/10.1007/978-3-319-97749-2_9

25. Yedroudj, M., Chaumont, M., Comby, F., Oulad Amara, A., & Bas, P. (2020). Pixels-off: Data-augmentation Complementary Solution for Deep-learning Steganalysis. *Proceedings of the 2020 ACM Workshop on Information Hiding and Multimedia Security* (pp. 39-48). https://doi.org/10.1145/3369412.3395061

26. Long, M., Wang, J., Ding, G., Pan, S. J., & Philip, S. Y. (2013).

Adaptation regularization: A general framework for transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 26(5), 1076-1089. https://doi.org/10.1109/TKDE.2013.111

27. Pevny, T., & Fridrich, J. (2007). Merging Markov and DCT features for multi-class JPEG steganalysis. *Security, Steganography, and Watermarking of Multimedia Contents IX* (Vol. 6505, p. 650503). International Society for Optics and Photonics. https://doi.org/10.1117/12.696774
28. Fridrich, J., & Kodovsky, J. (2012). Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3), 868-882. https://doi.org/10.1109/TIFS.2012.2190402
29. Kodovský, J., & Fridrich, J. (2009). Calibration revisited. *Proceedings of the 11th ACM Workshop on Multimedia and Security* (pp. 63-74). https://doi.org/10.1145/1597817.1597830
30. Holub, V., & Fridrich, J. (2015). Low-complexity features for JPEG steganalysis using undecimated DCT. *IEEE Transactions on Information Forensics and Security*, 10(2), 219-228.https://doi.org/10.1109/TIFS.2014.2364918
31. Lin, Y., Wang, R., Dong, L., Yan, D., & Wang, J. (2021). Tackling the cover source mismatch problem in audio steganalysis with unsupervised domain adaptation. *IEEE Signal Processing Letters*, 28, 1475-1479.
32. Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345-1359. https://doi.org/10.1109/TKDE.2009.191
33. Borgwardt, K. M., Gretton, A., Rasch, M. J., Kriegel, H. P., Schölkopf, B., & Smola, A. J. (2006). Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14), 49-57. https://doi.org/10.1093/bioinformatics/btl242
34. Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., & Darrell, T. (2014). Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv*:1412.3474.
35. Long, M., Cao, Y., Wang, J., & Jordan, M. (2015). Learning transferable features with deep adaptation networks. *International Conference on Machine Learning* (pp. 97-105).
36. Long, M., Zhu, H., Wang, J., & Jordan, M. I. (2017). Deep transfer learning with joint adaptation networks. *International Conference on Machine Learning* (pp. 2208-2217).
37. Long, M., Cao, Z., Wang, J., & Jordan, M. I. (2017). Conditional adversarial domain adaptation. *arXiv preprint arXiv*:1705.10667.
38. Pei, Z., Cao, Z., Long, M., & Wang, J. (2018). Multi-adversarial domain adaptation. *Thirty-second AAAI Conference on Artificial Intelligence*. (pp. 3934–3941).
39. Wang, J., Chen, Y., Yu, H., Huang, M., & Yang, Q. (2019). Easy transfer learning by exploiting intra-domain structures. *2019 IEEE International Conference on Multimedia and Expo (ICME)* (pp. 1210-1215). https://doi.org/10.1109/ICME.2019.00211
40. Zhu, Y., Zhuang, F., Wang, J., Ke, G., Chen, J., Bian, J., ... & He, Q. (2020). Deep subdomain adaptation network for image classification. *IEEE Transactions on Neural Networks and Learning Systems*, 32(4), 1713-1722. https://doi.org/10.1109/TNNLS.2020.2988928
41. Denemark, T., Sedighi, V., Holub, V., Cogranne, R., & Fridrich, J. (2014). Selection-channel-aware rich model for steganalysis of digital images. *2014 IEEE International Workshop on Information Forensics and Security (WIFS)* (pp. 48-53). https://doi.org/10.1109/WIFS.2014.7084302
42. Pasquet, J., Bringay, S., & Chaumont, M. (2014). Steganalysis with cover-source mismatch and a small learning database. *In 2014 22nd European Signal Processing Conference (EUSIPCO)* (pp. 2425-2429).
43. Bas, P., Filler, T., & Pevný, T. (2011). "Break our steganographic system": the ins and outs of organizing BOSS. *International Workshop on Information Hiding* (pp. 59-70). https://doi.org/10.1007/978-3-642-24178-9_5
44. Schaefer, G., & Stich, M. (2004). UCID: An uncompressed color image database. *Storage and Retrieval Methods and Applications for Multimedia 2004* (Vol. 5307, pp. 472-480).
45. Agustsson, E., & Timofte, R. (2017). Ntire 2017 challenge on single image super-resolution: Dataset and study. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (pp. 126-135).
46. Huiskes, M. J., & Lew, M. S. (2008). The MIR flickr retrieval evaluation. *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval* (pp. 39-43). https://doi.org/10.1145/1460096.1460104