

STEGANOGRAPHER DETECTION VIA DEEP RESIDUAL NETWORK

Mingjie Zheng¹, Sheng-hua Zhong^{1,*}, Songtao Wu^{1,2}, Jianmin Jiang¹

¹College of Computer Science and Software Engineering, Shenzhen University

²Department of Computing, The Hong Kong Polytechnic University

zhengmingjie@email.szu.edu.cn, {csshzhong, csstwu, jianmin.jiang}@szu.edu.cn

ABSTRACT

Steganographer detection problem is to identify culprit actors, who try to hide confidential information with steganography, among many innocent actors. This task has significant challenges, including various embedding steganographic algorithms and payloads, which are usually avoided in steganalysis under laboratory conditions. In this paper, we propose a novel steganographer detection model based on deep residual network. The proposed method strengthens the signal coming from secret messages, which is beneficial for the discrimination between guilty actors and innocent actors. Comprehensive experiments demonstrate that the proposed model achieves very low detection error rates in steganographer detection task. It also outperforms the classical rich model method and other CNN based method. Moreover, the model shows the robustness of inter-steganographic algorithms and inter-payloads.

Index Terms— Steganographer detection, steganalysis, deep residual network, convolutional neural network

1. INTRODUCTION

As a branch of information hiding, steganography aims to embed secret messages to digital media so that the very presence of hidden data is not obvious in covert communication [1]. The media with and without hidden information are called *stego* media and *cover* media, respectively. Steganalysis, on the other hand, strives to detect the

presence of embedded secrets in these digital media by the steganographer. Currently, most of traditional steganalysis techniques focus on classifying the cover images and stego images. This problem is called as the *stego detection* problem. In this paper, we are motivated to investigate a different problem: detecting which users are suspicious of using steganography to deliver hidden information among many innocent users [2]. This kind of user is termed as the guilty user (or guilty actor). This problem is termed as the *steganographer detection* problem. Compared with the stego detection problem, the steganographer detector is much harder. In such a case, the detector needs to see vast numbers of images, transmitted by a variety of users, and the embedding steganography and the corresponding amount of payload used by the guilty users are usually diverse. This requirement is completely different to the “laboratory conditions” found in most steganalysis experiments, and the challenges are different from classifying an individual image as cover or stego [3]. Security issues challenges are being amplified in this era [4]. In practice, we expect the proposed method for steganographer detection task to be more robust when the payload and steganographic algorithm are mismatched.

Existing methods for the steganographer detection problem mainly rely on two approaches: cluster analysis [2, 5] and anomaly detection [3, 6]. As the first attempt, in 2011, Ker *et al.* proposed a clustering paradigm for this problem [5]. They first extracted the features based on traditional steganalysis algorithms, *i.e.* PEV-274 [7], which is a combination of the extended DCT and calibrated Markov features. These features are then used to calculate the distances between each pair of actors by Maximum Mean Discrepancy (MMD) [8]. Finally, the agglomerative hierarchical clustering is used to find the guilty actor who deviates the most from the rest. Then, Ker *et al.* proposed a ranking based method based on Local Outlier Factor (LOF) [9] to reveal the guilty actor [3, 6]. Recently, Li *et al.* proposed a method that uses high-order joint features and clustering ensembles [2]. The proposed method is effective and efficient in identifying potential steganographers in large-scale social media networks.

In these years, deep learning based methods have achieved great success in many tasks, such as face

This work was supported by the National Natural Science Foundation of China (No. 61502311, No. 61620106008), the Natural Science Foundation of Guangdong Province (No. 2016A030310053), the Science and Technology Innovation Commission of Shenzhen under Grant (No. JCYJ20150324141711640), Special Program for Applied Research on Super Computation of the NSFC-Guangdong Joint Fund (the second phase), Shenzhen Emerging Industries of the Strategic Basic Research Project under Grant (No. JCYJ20160226191842793), the Shenzhen University research funding (201535), the Shenzhen high-level overseas talents program, and the Tencent “Rhinoceros Birds” - Scientific Research Foundation for Young Teachers of Shenzhen University.

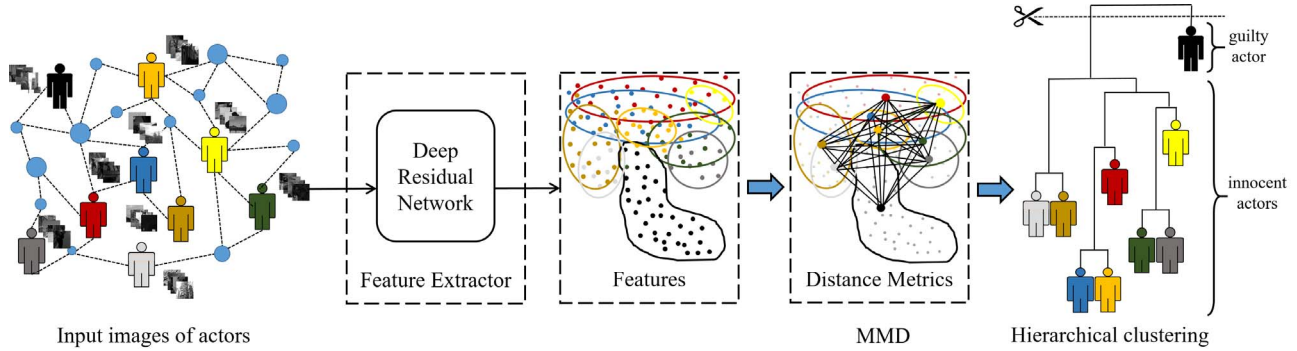


Fig. 1. Deep residual network based steganographer detection.

recognition [10] and image retrieval [11]. One of the promises of deep learning methods is to replace handcrafted features with efficient unsupervised or semi-supervised feature learning and hierarchical feature extraction [12]. Recent works for steganalysis also benefit from the progress in deep learning techniques [13-16]. As the first attempt to utilize the deep learning method, in 2014, Tan and Li proposed a stacked convolutional auto-encoder model for image steganalysis [13]. The proposed network shows better performance than the conventional steganalysis method based on subtractive pixel adjacency matrix [17], but it is still inferior to the performance of Spatial Rich Model (SRM) [18]. In 2015, Qian *et al.* proposed a novel customized Gaussian-Neuron Convolutional Neural Network for steganalysis [14]. The proposed model can capture the complex dependencies in images that are useful for steganalysis. Their model achieves comparable performance to SRM. Xu *et al.* reported a well-designed CNN architecture that took into account the knowledge of steganalysis [15]. Although the proposed model is neither large nor deep, compared with SRM, the performance of their model is competitive for attacking some steganographic algorithms. Recently, Wu *et al.* proposed a steganalysis method based on deep residual network [16]. The proposed model achieves significantly better performance than the classical rich model method [18] and other Convolutional Neural Network (CNN) based methods.

To our best knowledge, we are the first to propose a steganographer detection framework based on deep learning method. In our proposed method, the deep residual network is trained to discriminate the cover images and stego images. In the testing stage, the learnt model is used to extract distinguished features from images of each actor. Then, the agglomerative hierarchical clustering algorithm is used to cluster different actors based on their distance metrics. Finally, the actor who is deviated from other actors is identified as the guilty actor.

The rest of this paper is organized as follows. In Section 2, we propose a novel steganographer detection framework and underlying algorithm in detail. In Section 3, we provide a series of experiments to validate the proposed

method. Finally, the paper is closed with conclusion and future work.

2. PROPOSED METHOD

In this paper, we propose a novel steganographer detection method via deep residual network. Figure 1 illustrates our scheme of Residual Network based Steganographer Detection (RNSD). The residual neural network is utilized to extract features from each image of each actor. Then, the distance between each actor is calculated by MMD algorithm. Finally, we detect the guilty actor by the agglomerative hierarchical clustering algorithm.

2.1. Feature extraction based on deep residual network

As shown in Figure 2, the network consists of two sub-networks: the High-Pass Filtering (HPF) sub-network and the residual learning sub-network.

The HPF network filters the input image by the KV filter [14]. This high-pass filter is used to extract high frequency information from the input images. The reason is that the stego signal generated by message embedding is actually a high frequency signal.

In residual learning sub-network, there are two processing stages for feature learning. In order to generate enough statistical features for discriminating cover images and stego images, the network first use 64 convolutional kernels to preprocess the filtered images. The convolutional layer is followed by a batch normalization layer, a ReLU activation layer and a maximum pooling layer. Then, the network utilizes a series of residual learning (ResL) blocks and dimension increasing blocks to extract effective features for clustering. The configuration of the ResL block and the dimension increasing block is illustrated as Figure 2. In this paper, the ResL block we use consists of two convolutional layers, where each convolutional layer is followed by a batch normalization layer and a ReLU activation layer. The dimension increasing is similar to the ResL block, the only difference is that the number of output feature map is doubled and each feature map is down-sampled.

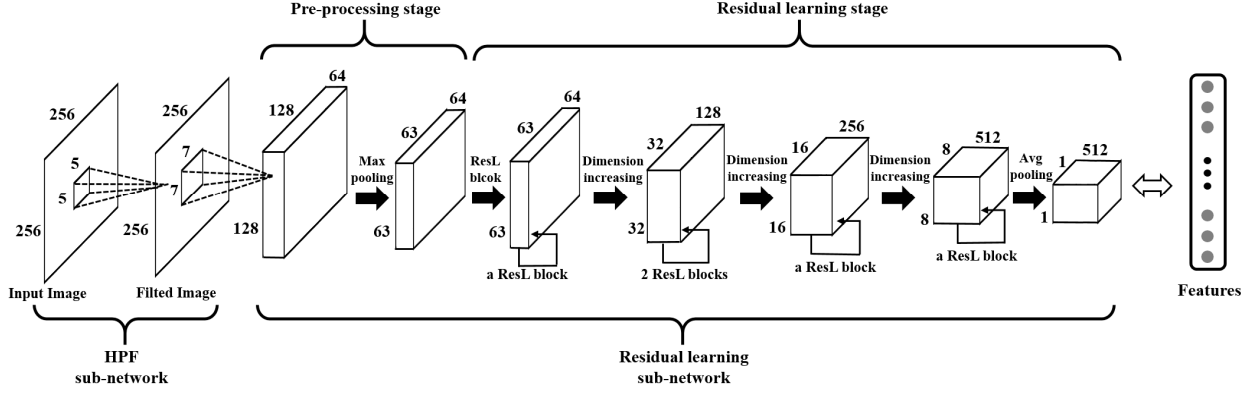


Fig. 2. Feature extraction based on deep residual network. The network has two sub-networks: the HPF sub-network and the residual learning sub-network. For HPF sub-network, it uses a fixed high-pass kernel to filter input cover images and stego images, aiming to suppress the image content. For the residual learning sub-network, it is to extract effective features for discriminating cover images and stego images. In our model, the number of ResL block is set to [1, 2, 1, 1] in the residual learning stage. Each ResL block is followed by a dimension increasing block. The last 512 feature maps we extracted can be features for steganographer detection.

For a building block (the ResL block or the dimension increasing block) in residual learning sub-network, instead of approximating an underlying function $H(\mathbf{z})$ directly, it fits the residual mapping $H(\mathbf{z})-\mathbf{z}$. For the stego image, it actually consists of the cover signal and the weak stego signal generated by message embedding. By feeding an image into a residual learning block, the cover signal is put forward to the output of the block, while the residual mapping fits the weak stego signal. Therefore, the weak stego signal can be emphasized through the whole network, which is suitable for the model to learn capture the weak signal information. For the steganographer detection, we take advantage of the residual learning block to capture the weak stego signal generated by the guilty actor.

To train the deep residual network, we use the fully-connected layers to classify the cover images and stego images, and this layer has 1,000 neurons. The model map the last 512 feature maps extracted from residual learning sub-network into binary labels. In the inference stage, we consider the model as feature extractor. The last layer of the residual learning sub-network produces a 512-dimensional (512-D) feature vector of each image for each actor.

2.2. Distance metrics calculation by MMD

We use Maximum Mean Discrepancy as the distance metrics to calculate the distance between each pair of actors. MMD is a measure of the similarity between two probability distributions, which can also be used as a measure to compare the features from two actors.

In this paper, we randomly selected m actors, which include one guilty actor and $m-1$ innocent actors, each of them transmits n images. Giving features $\mathbf{F}_X = (\mathbf{x}_1, \dots, \mathbf{x}_n)$ and $\mathbf{F}_Y = (\mathbf{y}_1, \dots, \mathbf{y}_n)$ obtained from actor X and Y , \mathbf{x}_i and \mathbf{y}_i ($1 \leq i \leq n$) represent a 512-D feature vector for an image from actor X and Y , respectively. The sample estimate for

the MMD distance of two actors $d(X, Y)$ is defined as follow:

$$d(X, Y) = \sqrt{\frac{1}{n^2} \sum_{i,j=1}^n k(\mathbf{x}_i, \mathbf{x}_j) - \frac{2}{n^2} \sum_{i,j=1}^n k(\mathbf{x}_i, \mathbf{y}_j) + \frac{1}{n^2} \sum_{i,j=1}^n k(\mathbf{y}_i, \mathbf{y}_j)} \quad (1)$$

where k is a bounded universal kernel that defines the dot product in the Reproducing Kernel Hilbert Space (RKHS). In our method, we apply the Gaussian kernel in Equation (2) to calculate the MMD distance.

$$k(\mathbf{x}, \mathbf{y}) = \exp(-\gamma \|\mathbf{x} - \mathbf{y}\|^2) \quad (2)$$

where γ is the inverse kernel width, which is set to the median of the L_2 -distances between features of two actors.

2.3. Steganographer detection via hierarchical clustering

Once the distance between each pair of actors is calculated based on MMD, we detect the guilty actor via the agglomerative hierarchical clustering algorithm.

The agglomerative hierarchical clustering algorithm groups objects over a variety of scales by creating a hierarchical tree. Initially, the similarity between each pair of actors is calculated based on MMD distance metrics. Next, each actor is firstly considered as a singleton cluster. The two nearest singleton clusters are linked using the linkage function, and form a new larger cluster. Then, it is repeated until all clusters have been linked and a complete binary hierarchical tree is formed. In the steganographer detection, all the innocent actors should be grouped into a single cluster and the other cluster only consists of the guilty actor.

In this paper, four versions of linkage functions are used for computing distance between clusters, including: the single linkage, the complete linkage, the average linkage and the weighted average linkage. The single linkage uses the smallest distance between actors in the two clusters to represent the similarity between clusters. On the contrary, the complete linkage defines the similarity based on the

largest distance between actors in the two clusters. The average distance between all pairs of actors in two clusters is applied for the average linkages. The weighted average linkage adopts a recursive definition for the distance between two clusters. If cluster \mathbf{r} is created by combining clusters \mathbf{p} and \mathbf{q} , and the distance $D(\mathbf{r}, \mathbf{s})$ between \mathbf{r} and another cluster \mathbf{s} is defined as follow:

$$D(\mathbf{r}, \mathbf{s}) = \frac{D(\mathbf{p}, \mathbf{s}) + D(\mathbf{q}, \mathbf{s})}{2} \quad (3)$$

where $D(\mathbf{p}, \mathbf{s})$ is the distance between clusters \mathbf{p} and \mathbf{s} and $D(\mathbf{q}, \mathbf{s})$ is the distance between clusters \mathbf{q} and \mathbf{s} .

3. EXPERIMENTS

3.1. Experimental setting

In the experiments, the image source used is the BOSSbase ver 1.01 [19]. The original BOSSbase contains 10,000 grayscale natural images with the size of 512×512. Following the setting in [14, 16], each image in the dataset is cropped into 4 non-overlapping 256×256 in our experiments. We randomly select 20,000 cover images from the cropped BOSSbase, and their stegos which are generated by Spatial version of the UNiVersal Wavelet Relative Distortion (S-UNIWARD) steganography [20] at payload 0.4 bit-per-pixel (bpp). These 40,000 images are used for training the residual network. The rest 20,000 covers as the transferring images for validating steganographer detection. In our experiments, we randomly select m actors, each of whom transmit n images. Each time, m is set to 100 and n is set to 200. All the statistical experiments are repeated for ten times, and the average results are reported.

Five steganographic algorithms are used for validation, including: Highly Undetectable steGO implemented using the Gibbs construction with Bounding Distortion (HUGO-BD) [21], Wavelet Obtained Weights (WOW) [22], S-UNIWARD [20], HIgh-pass Low-pass Low-pass (HILL) [23], and Minimizing the power of the most POwerful Detector (MiPOD) [24].

We compare the proposed method with two other methods: SRMQ1_SD and ANSD. SRMQ1_SD is the abbreviation of the steganographer detection method via SRMQ1 [18], which is the spatial rich model with a single quantization step. ANSD is the abbreviation of the steganographer detection method based on a well-known deep convolutional neural network architecture AlexNet [25]. To the parameters such as the learning rate for AlexNet, we follow the general setting in [25].

3.2. A single steganographic algorithm with one payload

In the first experiment, we try to compare the performance of the proposed method with the classical method SRMQ1_SD, and the CNN-based method ANSD. We use the single linkage as the clustering linkage method. For the guilty actor, two hundred stego images are embedded by S-

Table 1. The comparison results of different methods on S-UNIWARD with 0.4bpp.

Method	Feature Dimension	Average Distance		Acc. (%)	STD
		AD1	AD2		
RNSD	512	0.078	0.349	100	0
ANSD	1,000	0.076	0.275	100	0
SRMQ1_SD	12,753	0.076	0.085	10	0.32

UNIWARD with 0.4bpp. The comparison results are shown in Table 1.

In Table 1, we can find two methods based on convolutional neural networks can detect the guilty actor accurately. However, the accuracy of SRMQ1 is only 10%. In addition, the feature dimension for SRMQ1 is 12,573, which is much larger than other two methods. AD1 is the average distance between the features from innocent users. AD2 is the average distance between the feature from innocent user and the guilty user. As we expect, AD1 is smaller than AD2 in all cases. Compared with the ratio of AD1/AD2 for SRMQ1_SD, this value for RNSD and ANSD is much smaller. It means the features extracted by CNN-based methods from the guilty actor are deviated from those of innocent actors. Based on these results, we can conclude that the CNN-based steganographer detection methods have demonstrated superior performance than traditional rich model SRMQ1_SD. Unlike SRMQ1 that utilizes handcrafted features, CNN-based methods have the ability to directly capture effective features from the various correlations in images.

3.3. Multiple steganographic algorithms with a single payload

In this experiment, we test the performance of the proposed model when the stego images are embedded by five different steganographic algorithms with a single payload. All models are trained based on S-UNIWARD at 0.4bpp, but tested on multiple steganographic algorithms, including: HUGO-BD, WOW, S-UNIWARD, HILL, and MiPOD. For the guilty actor, two hundred stego images can be equally divided into five groups, and each 40 images are embedded by each steganography. The detection accuracy comparison results are provided in Table 2. The payload is set to 0.05, 0.1, 0.2, 0.3, 0.4, and 0.5, respectively. We also use the single linkage as the clustering linkage method.

As Table 2 shows, in each case, the detection accuracy of our model is 100%. When the payload is equal to 0.5, the detection accuracy of SRMQ1_SD is 30%. The detection accuracy of SRMQ1_SD decreases with the decrease of the payload. Consistent with our expectation, a similar situation also happens in ANSD. When the payload is greater than or equal to 0.2, ANSD can also achieve 100% detection accuracy. But when the payload decreases to 0.1, the accuracy of ANSD drops to 20%. To the payload is 0.05, ANSD cannot achieve successful detection in ten times. As

Table 2. The detection accuracy comparisons of multiple steganographic algorithms with a single payload, the payload ranges from 0.05 to 0.5.

Payload (bpp)	Method	Acc. (%)	STD
0.05	RNSD	100	0
	ANSD	0	0
	SRMQ1_SD	0	0
0.1	RNSD	100	0
	ANSD	20	0.42
	SRMQ1_SD	0	0
0.2	RNSD	100	0
	ANSD	100	0
	SRMQ1_SD	0	0
0.3	RNSD	100	0
	ANSD	100	0
	SRMQ1_SD	0	0
0.4	RNSD	100	0
	ANSD	100	0
	SRMQ1_SD	10	0.32
0.5	RNSD	100	0
	ANSD	100	0
	SRMQ1_SD	30	0.48

we known, when the embedded payload is lower, the stego image is more similar to the cover image. It means it is harder to distinguish and identify the guilty actor from the innocent actors. But in this case, our proposed method has the ability to identify the guilty actor accurately.

We also explore the effects of different versions of linkages as the clustering linkage methods in Figure 3. We test RNSD and ANSD with different linkages when the payload is 0.1bpp. With different linkages, the proposed method can achieve 100% detection accuracy.

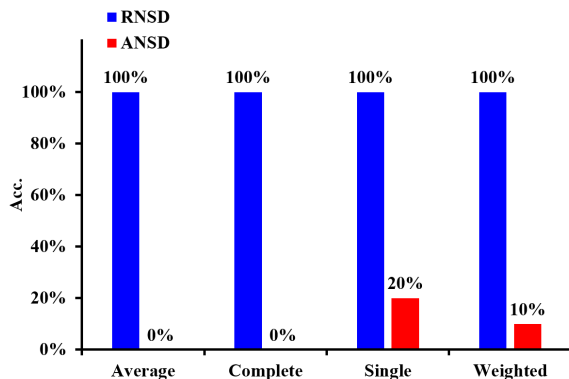


Fig. 3. The detection accuracies of the proposed method RNSD and ANSD with different linkages when the payload is 0.1bpp.

3.4. Performance comparisons of different clustering methods

In this paper, we use the agglomerative hierarchical clustering algorithm (HC) to cluster and identify the guilty actor. This clustering algorithm is a very simple one. We

expect that a more effective clustering method could achieve better performance in steganographer detection. Thus, we use another two clustering algorithms to do the empirical evaluations, including Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [26] and High Density Peaks Search (HDPS) [27]. DBSCAN groups the points that are closely packed together (the points with many nearby neighbors), and marking the points as outliers that lie alone in low-density regions (whose nearest neighbors are too far away). HDPS is based on the idea that cluster centers are characterized by a higher density than their neighbors and by a relatively large distance from points with higher densities [27].

In this experiment, we generate the stego images by multiple steganographic algorithms with multiple payloads. That is, for the guilty actor, two hundred stego images are equally divided into 25 groups. The images from each group are embedded by a single steganographic algorithm with a single payload. HUGO-BD, WOW, S-UNIWARD, HILL and MiPOD are utilized as the steganographic algorithms. And we use the payload including: 0.05, 0.1, 0.2, 0.3, and 0.4bpp, respectively. The detection accuracies based on two clustering algorithms are shown in Table 3. We use the average linkage as the clustering linkage method in HC model.

Table 3. The detection accuracies based on different clustering algorithms.

Feature extraction methods	Clustering algorithms					
	HC		DBSCAN		HDPS	
	Acc. (%)	STD	Acc. (%)	STD	Acc. (%)	STD
Residual Net	100	0	100	0	100	0
AlexNet	90	0.32	90	0.32	100	0
SRMQ1	0	0	0	0	0	0

From Table 3, we can observe that deep learning based methods can detect the guilty actor more accurately. Moreover, we can find the accuracy of AlexNet+HDPS is 100% and the accuracy of AlexNet+DBSCAN is 90%. Therefore, with a more effective clustering algorithm, deep learning based method can have a better performance on steganographer detection problem. These findings prove that deep learning based methods have potentials to handle the steganographer detection problem.

4. CONCLUSION AND FUTURE WORKS

In this paper, we propose a novel steganographer detection model based on deep residual network. To our best knowledge, we are the first to propose a steganographer detection framework based on deep learning method. In our method, deep residual network is trained to distinguish the cover images and the stego images with embedded weak signal. Then the learnt model is used to extract distinguished features from images of each actor. The agglomerative hierarchical clustering algorithm is introduced to find the

guilty actor based on distance metrics, who deviates the most from the rest actors.

In the experiments, we test the performance of the proposed method with the classical method and the CNN-based method. Our proposed method trained with the cover images and the stego images embedded by a single steganographic algorithm method with a single payload could detect the guilty actor accurately from easy to hard conditions. Moreover, the model demonstrates the robustness of inter-steganographic algorithms and inter-payloads. Our first future work is to expand our method in the compressed domain. Another meaningful future work is to apply the proposed method into the steganographer detection problem and investigate the case of multiple guilty actors in the large-scale social networks.

5. REFERENCES

- [1] R. J. Anderson and F. A. P. Petitcolas, "On the limits of steganography," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 4, pp. 474–481, May 1998.
- [2] F. Li, K. Wu, J. Lei, M. Wen, Z. Bi and C. Gu, "Steganalysis Over Large-Scale Social Networks With High-Order Joint Features and Clustering Ensembles," *TIFS*, vol. 11, no. 2, pp. 344–357, Feb. 2016.
- [3] A. D. Ker and T. Pevný, "The steganographer is the outlier: Realistic large-scale steganalysis," *TIFS*, vol. 9, no. 9, pp. 1424–1435, Sep. 2014.
- [4] K. Gai, M. Qiu, Z. Ming, H. Zhao and L. Qiu, "Spoofing-Jamming Attack Strategy Using Optimal Power Distributions in Wireless Smart Grid Networks," *TSG*, Feb. 2017.
- [5] A. D. Ker and T. Pevný, "A new paradigm for steganalysis via clustering," in *Proc. SPIE, Media Watermark., Security, Forensics III*, vol. 7880, pp. U01–U13, Feb. 2011.
- [6] A. D. Ker and T. Pevný, "Identifying a steganographer in realistic and heterogeneous data sets," in *Proc. SPIE, Media Watermark., Security, Forensics XIV*, vol. 8303, pp. N01–N13, May 2012.
- [7] T. Pevný and J. Fridrich, "Merging Markov and DCT features for multi-class JPEG steganalysis," in *Proc. SPIE, Media Watermark., Security, Forensics IX*, vol. 6505, pp. 3–14, Feb. 2007.
- [8] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample-problem," in *Proc. NIPS*, 2007, pp. 513–520.
- [9] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2000, pp. 93–104.
- [10] Z. Wu and W. Deng, "One-shot deep neural network for pose and illumination normalization face recognition," in *Proc. ICME*, Seattle, 2016, pp. 1–6.
- [11] V. A. Nguyen and M. N. Do, "Deep learning based supervised hashing for efficient image retrieval," in *Proc. ICME*, Seattle, 2016, pp. 1–6.
- [12] H. A. Song and S. Y. Lee, "Hierarchical representation using NMF," in *Proc. ICONIP*, Springer Berlin Heidelberg, 2013, pp. 466–473.
- [13] S. Tan and B. Li, "Stacked convolutional auto-encoders for steganalysis of digital images," in *Proc. APSIPA*, Siem Reap, 2014, pp. 1–4.
- [14] Y. Qian, J. Dong, W. Wang, and T. Tan, "Deep learning for steganalysis via convolutional neural networks," in *Proc. SPIE Media Watermark. Security, Forensics*, 2015, vol. 9409, pp. J-1–J-10.
- [15] G. Xu, H. Z. Wu and Y. Q. Shi, "Structural Design of Convolutional Neural Networks for Steganalysis," *SPL*, vol. 23, no. 5, pp. 708–712, May 2016.
- [16] S. Wu, S. Zhong and Y. Liu, "Deep residual learning for image steganalysis," *MTAP*, pp. 1–17, 2017.
- [17] T. Pevný, P. Bas and J. Fridrich, "Steganalysis by Subtractive Pixel Adjacency Matrix," *TIFS*, vol. 5, no. 2, pp. 215–224, Jun. 2010.
- [18] J. Fridrich and J. Kodovsky, "Rich Models for Steganalysis of Digital Images," *TIFS*, vol. 7, no. 3, pp. 868–882, Jun. 2012.
- [19] P. Bas, T. Filler, and T. Pevný, "Break our steganographic system—The ins and outs of organizing BOSS," in *Proc. Information Hiding*, May 2011, vol. 6958, pp. 59–70.
- [20] V. Holub and J. Fridrich, "Universal distortion design for steganography in an arbitrary domain," *EURASIP Journal on Information Security*, vol. 2014, no. 1, pp. 1–13, 2014.
- [21] T. Filler and J. Fridrich, "Gibbs Construction in Steganography," *TIFS*, vol. 5, no. 4, pp. 705–720, Dec. 2010.
- [22] V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," in *Proc. WIFS*, Tenerife, 2012, pp. 234–239.
- [23] B. Li, M. Wang, J. Huang and X. Li, "A new cost function for spatial image steganography," in *Proc. ICIP*, Paris, 2014, pp. 4206–4210.
- [24] V. Sedighi, R. Coganne and J. Fridrich, "Content-Adaptive Steganography by Minimizing Statistical Detectability," *TIFS*, vol. 11, no. 2, pp. 221–234, Feb. 2016.
- [25] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proc. NIPS*, 2012.
- [26] M. Ester, H. Kriegel, J. Sander, X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise", in *Proc. KDD*, vol. 96, pp. 226–231, Aug. 1996.
- [27] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, Jun. 2014.