Estimation of the Hidden Message Length in Steganography: A Deep Learning Approach

François Kasséné Gomis, Thierry Bouwmans, Mamadou Samba Camara, Idy Diop

Estimation of the Hidden Message Length in Steganography: A Deep Learning Approach

François Kasséné Gomis[1], Thierry Bouwmans[2], Mamadou Samba Camara[1], and Idy Diop[1]

[1] Cheikh Anta Diop University, Sénégal
gomisfk@gmail.com
[2] La Rochelle univ., France
thierry.bouwmans@univ-lr.fr

Abstract. Steganography is a science which helps to hide secret data inside multimedia supports like image, audio and video files to ensure secure communication between two parts of a channel. Steganalysis is the discipline which detects the presence of data hidden by a steganographic algorithm. There are two types of steganalysis: targeted steganalysis and universal steganalysis. In targeted steganalysis, the steganographic algorithm used to hide data is known. In the case of universal steganalysis, the detection of hidden data doesn't depend on any specific algorithm used in the process of steganography. In this paper, we focus on universal steganalysis of images in a database with an eventual cover-source mismatch problem. It is shown that combining both unsupervised and supervised machine learning algorithms helps to improve the performance of classifiers in the case of universal steganalysis by reducing the cover-source mismatch problem. In the unsupervised step, the k-means algorithm is generally used to group similar images. When the number of features extracted from the image is very large it becomes difficult to compute the k-means algorithm properly. We propose, in that case, to use Deep Learning with Convolutional Neural Network (CNN) to group similar images at first and implement a Multilayer Perceptron (MLP) neural network to estimate the hidden message length in all the different groups of images. The first step of this approach prevents the cover-source mismatch problem. Reducing this issue boost the performance of classifiers in the second step which consists of estimating the hidden message length.

Keywords: Steganography · Steganalysis · Machine Learning · Deep Learning · Convolutional Neural Networks · MultiLayer Perceptron.

1 Introduction

Research in universal steganalysis domain become very interesting since researchers discover that deep learning with Convolutional Neural Networks (CNNs) helps to obtain better results in the classification between cover and stego images. Till now CNNs have been never used for regression to estimate the hidden

message length. Various types of materials used to capture images and various steganographic algorithms available for hiding data cause a problem called cover-source mismatch in universal steganalysis. In previous studies, it is demonstrated that clustering can be used, as a prior step in the process of steganalysis, to improve the performance of the classifiers in a database with cover-source mismatch [9], before implementing a classification or regression algorithm for universal steganalysis. Generally, authors used clustering with the k-means algorithm to group images into clusters. However, if the number of features extracted is big, it becomes computationally difficult to compute them with the k-means algorithm. In this context, we propose to employ a deep learning-based approach for estimation of the hidden message length in steganography. We called the proposed method DeepStego. To estimate the hidden message length, we use in the first step a CNN for grouping similar images into different categories. Then, in the second step, we implement an MLP neural network to estimate the hidden message length.

The rest of the paper is organized as follows: In Section 2, we give the theory in universal steganalysis. In Section 3, we present our original method of universal steganalysis. Then, we illustrate our scheme in Section 4. In Section 5, experiments are conducted on a database and we discuss the results. Concluding remarks and future directions are provided in Section 6.

2 Related Works

Research in the universal steganalysis domain focuses either on the extraction of relevant features which are sensitive to any steganographic algorithm or in the machine learning algorithms used to build models for classification or regression. The goal in both cases is to help to boost the performance of classifiers. About relevant features for universal steganalysis of JPEG images, authors use First-order statistics, Inter-block, and Intra-block features. Table 1 is a summary of different categories of features and some authors who proposed them for universal steganalysis of images.

Table 1. Features for Universal Steganalysis: An Overview.

Authors	Categories	Feature Names
Ashu and Chhikara (2014) [1]	First-order statistics	Global histogram
		AC histograms
		Dual histograms
Chen and Shi (2008) [4]	Inter-block features	Co-occurrence matrix
		Variation
		Blockiness
Chen and Shi (2008) [4]	Intra-block features	Average Markov matrix

All these features are sensitive to steganographic algorithms embedding impact while at the same time insensitive to the image content. According to the

steganographic algorithm used to embed messages, a category of features can be more useful than others. Thus, some authors proposed methods that combine different categories of features [4]. Many different blind steganalysis methods have been proposed in the literature [8]. After choosing a set of features, we need to find a strong algorithm for binary classification or regression (to separate stego and cover images or to estimate the hidden message length). Support Vector Machine [6] and classical Neural Networks are very used for classification between stego and cover images. Recently, some authors start to use Convolutional Neural Network for the same task [5]. To estimate the relative payload, Multiple Linear Regression is also used but the cover-source mismatch problem and the huge number of features extracted make its implementation difficult. It is shown that applying clustering is a good solution before using it [9]. About clustering when the number of features extracted is huge and when the database is big (more than 10,000 images), the computation becomes difficult. Some papers related to deep learning for universal steganalysis have been published. Chaumont and al. made a recapitulation of those methods in their paper [3]. Deep learning with CNN has better performance than usual machine learning algorithms. However, to estimate the hidden message length in the case of universal steganalysis, there are still some difficult challenges to overcome to boost the performance of blind steganalyzers. Some methods which deal with estimation of hidden message length have been proposed in the literature [11]. CNN is a classification algorithm that has never been used in the perspective of estimating the hidden message length.

3 DeepStego: A Deep Learning Methodology

In this section, we detail the proposed approach named DeepStego for universal steganalysis in a database with a cover-source mismatch problem. Figure 1 shows an overview of the data pipeline. The different steps are described and illustrated in the following.

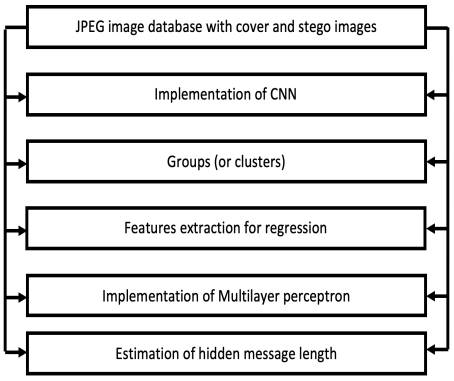

Fig. 1. DeepStego (Proposed method): Data Pipeline.

The proposed approach consists of combining deep learning with CNNs [10] and MLP neural network [6] to build strong and robust models that estimate the hidden message length in universal steganalysis. This method consists of three main steps:

- **Step 1:** Implementation of CNN in a JPEG database containing cover and stego images. The objective here is to group similar images into different groups (or clusters). Grouping images is a strategy to prevent an eventual cover-source mismatch problem in the database. That problem can occur when there is a variety of materials used to get images and a lot of different steganographic algorithms used to hide data into images.
- **Step 2:** Implementation of MLP in all the groups to build models for estimating the hidden message length.
- **Step 3:** Utilization of the models for prediction.

4 Experimental Illustrations

4.1 Cover and stego images

We use a steganography Python module called Stegano [2] to generate stego images with different payloads. As shown in Figure 2, after the embedding process, changes between stego and cover images are not visually detectable. Histograms of the cover image and its stego image are generated by the Stegano module algorithm.

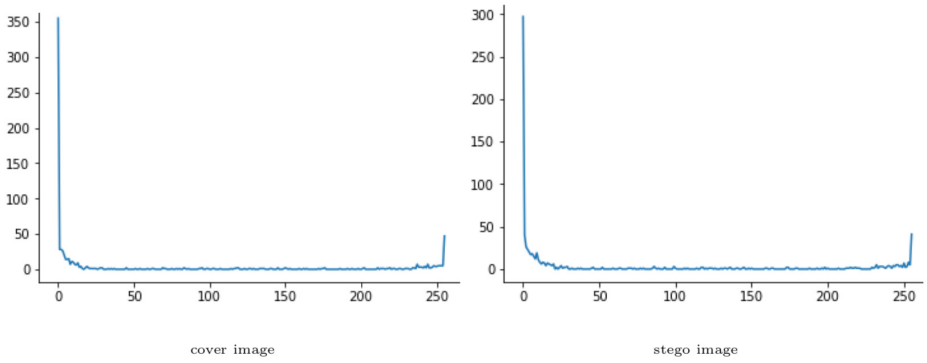

Fig. 2. Histograms comparison between the cover image (left) and the stego image (right).

4.2 The hidden message estimation technique

Image database description To illustrate our purpose, we use the MNIST Database [7]. This database is very practical in our case. It contains $70,000$ 28×28 grayscale JPEG images divided into 10 categories.

Stego images generation To generate stego images for simulation, we use the module Stegano of Python to embed messages with different lengths inside images. So, we obtain $35,000$ stego images. After that process, we create a vector of labels which contains the lengths of the hidden messages of all the images of the database. That vector will be used for the regression part.

Outcome vector L for CNN

$$\begin{pmatrix} L_1 \\ L_2 \\ . \\ . \\ L_{n-1} \\ L_n \end{pmatrix}$$

Outcome vector Y for MLP

$$\begin{pmatrix} Y_1 \\ Y_2 \\ . \\ . \\ Y_{n-1} \\ Y_n \end{pmatrix}$$

The vector L will be used for classification with CNN and the vector Y for regression with MLP.

Convolutional Neural Networks in the database To group images of the MNIST database into different categories, we implement a Convolutional Neural Network classifier with four hidden layers. This is a practical and very convenient database to highlight the proposed method. We perform CNN on the data before using Multilayer Perceptron for regression in the different groups. This architecture of CNN gives a good classification of the images into 10 groups.

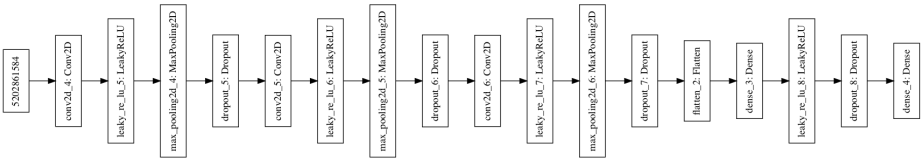

Fig. 3. CNN architecture on MNIST dataset.

This architecture can be changed. It depends on the database we use to perform universal steganalysis of images. The goal in this step is to reduce an eventual cover-source mismatch issue.

Features for regression with multilayer percetron (MLP) in the different categories of images To implement an MLP neural network, we use both intra-block and inter-block correlations [4]. It consists of 486 features extracted from a JPEG image. At this step, we need the labels (vector Y) containing the lengths of the embedded messages of all images in the database.

Steganalysis on a category of images To perform universal steganalysis on clusters, we use regression with an MLP neural network architecture with the most relevant features from both intra-block and inter-block correlations.

5 Experimental Results

Estimating the hidden message length is not an easy task. In the case of universal steganalysis combining CNN and MLP neural networks is a good approach to perform that task. In our experiments on the MNIST database, we got interesting results. This database is very convenient to illustrate our method of doing universal steganalysis in a database with a cover-source mismatch problem.

5.1 Deep learning with CNN for classification

By applying CNN with a standard architecture, we obtain easily 10 groups of images. Here an illustration of the model performance.

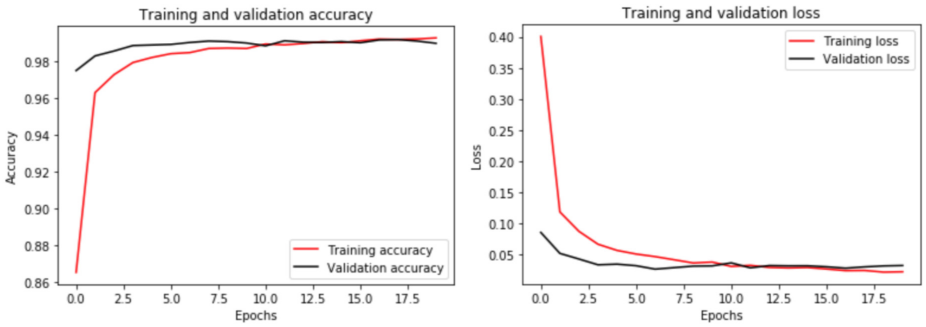

Fig. 4. Accuracy and loss in training and validation datasets.

In Figure 4, we can observe the evolution of the accuracy score in the training and validation data. We can note that they are very close.

5.2 MLP neural network for regression

In this step, we implement in all the clusters an MLP neural network for estimating the lengths of the hidden messages. Here the architecture of our neural network which consists of an input layer of 12 nodes (12 features selected from the 486 extracted features), three hidden layers of 13 nodes and an output layer of 1 node (estimation of the hidden message length)

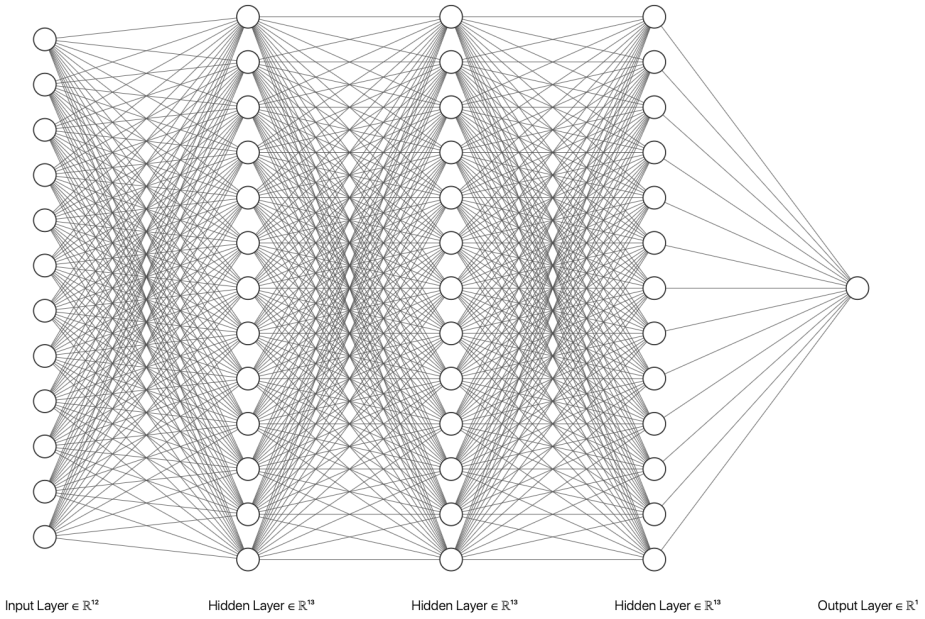

Fig. 5. MLP architecture employed in DeepStego

Table 2 shows the MLP models accuracy scores in the groups generated by CNN.

Table 2. Accuracy score value in the 10 groups for DeepStego.

Groups	accuracy score
Group 1	0.99
Group 2	0.99
Group 3	0.98
Group 4	0.99
Group 5	0.98
Group 6	0.98
Group 7	0.99
Group 8	0.98
Group 9	0.99
Group 10	0.99

Furthermore, the use of a stepwise feature selection helps to boost the MLP accuracy in the regression step.

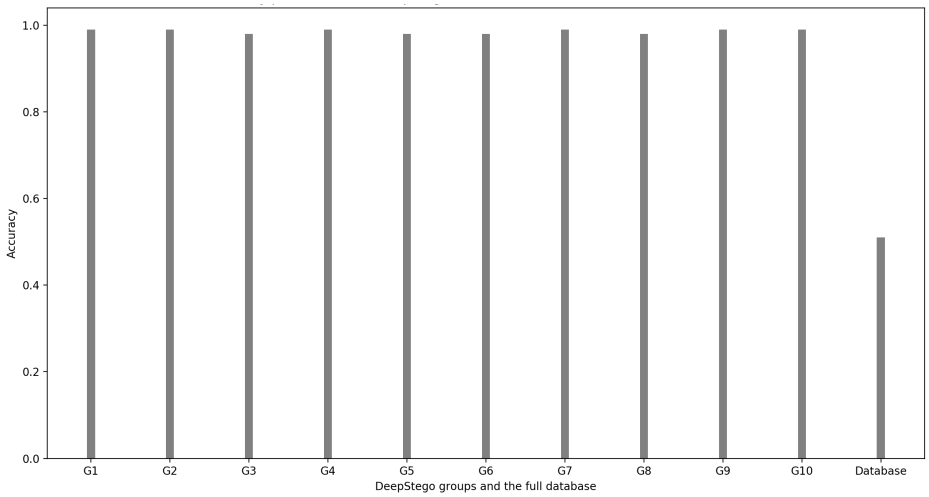

Fig. 6. Performance comparison between DeepStego (G1 to G10) and a normal universal steganalysis procedure (average score on the MNIST database).

A normal universal steganalysis procedure consists of extracting relevant features and implementing a supervised algorithm for classification or regression. For that, we implement an MLP neural network (with the same architecture implemented in the second part of DeepStego) to estimate the hidden message length on the full database MNIST. In Figure 6, we show a comparison between the results of a normal universal steganalysis procedure and DeepStego. The average score on the MNIST database for the universal steganalysis procedure is inferior to each score obtained by DeepStego in all the 10 groups. Thus, DeepStego gives better results (in all the 10 groups) than a universal steganalysis procedure on the full database.
However, the highest accuracies of the universal steganalysis approaches proposed in the literature, are often in the range [0.95, 0.97]. It rarely reaches 0.9. With DeepStego, we get accuracy which turns around 0.9 in all groups showing the interest of the proposed deep learning approach for the estimation of the hidden message length in steganography.

6 Conclusion

In this paper, we addressed the cover-source mismatch problem that prevents the utilization of regression for universal image steganalysis. For this, we need to group similar images into clusters before applying it. When the extracted feature vector from the image is very large, the k-means algorithm cannot help to perform the clustering process. To address this issue, we have proposed an original method that used in its first step CNNs to group similar images and in its second step implementation of a multilayer perceptron neural network

to estimate the hidden message length. Experimental results on the MNIST database provided good approximation models in all the 10 clusters. Thus, deep learning with CNN is a suitable alternative to k-means to reduce the cover-source mismatch problem in the case of universal steganalysis.

References

1. Ashu, A., Chhikara, R.: Performance evaluation of first and second order features for steganalysis. International Journal of Computer Applications **92** (03 2014). https://doi.org/10.5120/16093-5372
2. Bonhomme, C., al.: Stegano: a pure python steganography module (2010–), https://pypi.org/project/Stegano/, [Online; accessed October 16, 2019]
3. Chaumont, M.: Deep learning in steganography and steganalysis since 2015 (10 2018). https://doi.org/10.13140/RG.2.2.25683.22567
4. Chen, C., Shi, Y.Q.: JPEG image steganalysis utilizing both intrablock and interblock correlations. In: 2008 IEEE International Symposium on Circuits and Systems. pp. 3029–3032 (May 2008)
5. Chen, M., Sedighi, V., Boroumand, M., Fridrich, J.: JPEG-Phase-Aware Convolutional Neural Network for Steganalysis of JPEG Images. In: Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security. pp. 75–84. IH&MMSec '17, ACM, New York, NY, USA (2017)
6. Cortez, P.: Data Mining with Multilayer Perceptrons and Support Vector Machines, pp. 9–25. Springer Berlin Heidelberg, Berlin, Heidelberg (2012)
7. Deng, L.: The mnist database of handwritten digit images for machine learning research [best of the web]. IEEE Signal Processing Magazine **29**(6), 141–142 (Nov 2012)
8. Dwivedi, Y.P., Bera, M.S., Sharma, M.M.: Universal steganalysis techniques based on the feature extraction in transform domain (2017)
9. Gomis, F.K., Camara, M.S., Diop, I., Farssi, S.M., Tall, K., Diouf, B.: Multiple linear regression for universal steganalysis of images. In: 2018 International Conference on Intelligent Systems and Computer Vision (ISCV). pp. 1–4 (April 2018)
10. O'Shea, K., Nash, R.: An introduction to convolutional neural networks. ArXiv e-prints (11 2015)
11. Quach, T.T.: Extracting hidden messages in steganographic images. Digital Investigation **11**, S40S45 (08 2014). https://doi.org/10.1016/j.diin.2014.05.003