IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 8, 2019, accepted May 10, 2019, date of publication May 20, 2019, date of current version June 7, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2918086

Deep Learning Applied to Steganalysis of Digital Images: A Systematic Review

TABARES-SOTO REINEL[1], RAMOS-POLLÁN RAÚL[2], AND ISAZA GUSTAVO[3]

[1]Departamento de Electrónica y Automatización-Antigua Estación del Ferrocarril, Universidad Autónoma de Manizales, Manizales 170001, Colombia
[2]Departamento de Ingeniería de Sistemas, Universidad de Antioquia, Medellín 050001, Colombia
[3]Departamento de Sistemas e Informática, Universidad de Caldas, Manizales 170001, Colombia

Corresponding author: Tabares-Soto Reinel (rtabares@autonoma.edu.co)

This research is funded by the Universidad Autonóma de Manizales under the project entitled "Application of Convolutional Neural Networks for Steganalysis" with grant code 589-089.

ABSTRACT Steganography consists of hiding messages inside some object known as a carrier in order to establish a covert communication channel so that the act of communication itself goes unnoticed by observers who have access to that channel. The steganalysis is dedicated to the detection of hidden messages using steganography; these messages can be implicit in different types of media, such as digital images, video files, audio files or plain text. Traditionally, steganalysis has been divided into two separate stages, the first stage consists of manual extraction of sophisticated features and the second stage is classification using Ensemble Classifiers or Support Vector Machines. In recent years, the development of Deep Learning has made it possible to unify and automate the two traditional stages into an end to end approach with promising results. This paper shows the evolution of steganalysis in recent years using the Deep Learning techniques. The results of these techniques have surpassed those obtained with conventional methods - *Rich Models with Ensemble Classifiers* - both in the spatial and frequency (JPEG) domains. Since 2014, researchers have used The Convolutional Neural Networks to solve this problem generating diverse architectures and strategies to improve the detection percentages of steganographic images on the last generation algorithms (WOW, S-UNIWARD, HUGO, J-UNIWARD, among others). The Deep Learning, being applied to steganalysis, is now in the process of construction and results so far are encouraging for researchers that are interested in the topic.

INDEX TERMS Convolutional neural network, deep learning, steganalysis, steganography.

I. INTRODUCTION

Steganography consists of hiding messages inside digital multimedia files (images, sound, and video) imperceptibly for any receiver. The first documents describing the use of these techniques date back to the times of Herodotus in ancient Greece. One story describes how they sent a message to Sparta to warn that Xerxes intended to invade Greece so that it would be hidden from inspection and not arouse suspicion. At that time it was written on boards covered with wax. So, to camouflage the message they wrote directly on the wood covered it with wax and wrote on it again. At first glance one could only see the writing on the wax but, if it was removed, one could read the message hidden in the wood. During the Second World War, the most commonly used system was to microfilm a message and reduce it to the extreme of a small dot, so that it could pass as a punctuation mark of a character within another text. For example, the dot on the vowel "i" could be microfilm with a message [1]. This technique has become an exciting alternative to hide information because cryptography is not allowed in all countries [2]. The formulation of the steganography process is due to the famous Simmons Prisoner Problem explained in [3], which consists of two prisoners, Alice and Bob, who wish to exchange messages while being intercepted continuously by the prison director, Eve. If Eve considers the messages exchanged between Alice and Bob are suspicious, she will not allow the messages to reach the recipient.

Industrial steganography has been used to control the copying of digital material illegally, so copyright societies introduce information by modifying digital content in an imperceptible way to the human eye, with the aim of providing evidence of who owns the image or to whom it has been sold or sent [4]. At a military level, this technique has been

The associate editor coordinating the review of this manuscript and approving it for publication was Avishek Guha.

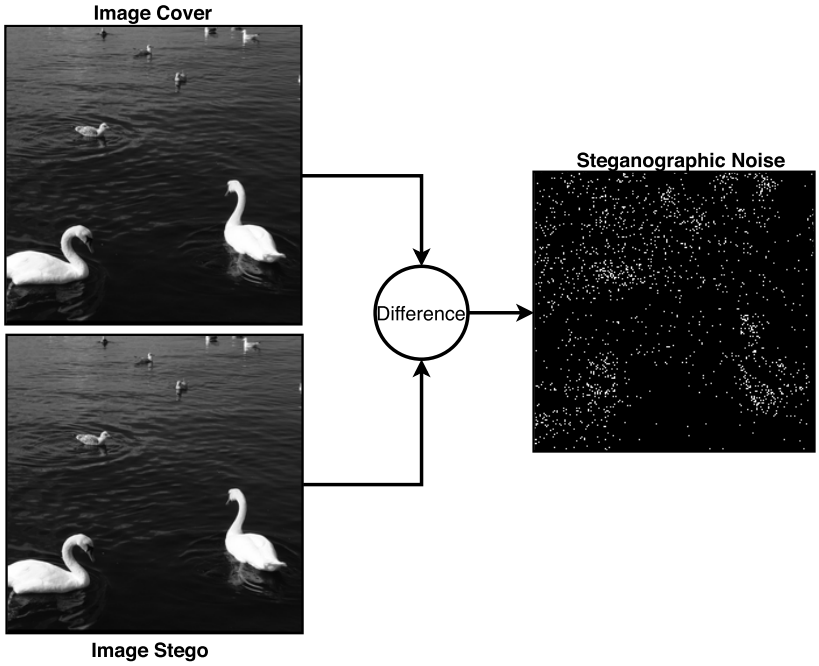

FIGURE 1. Example of embedding a message with the S-UNIWARD algorithm using a payload of. 4 bpp. Image taken from BOSSBase V1.01 [12].

used to transmit important messages without being identified by the counterpart. It is also believed that steganography could have been used even in communications of illegal groups and terrorists [4].

Steganography can be done from two domains: spatial or frequency. From the spatial domain, the algorithms are characterized by directly changing some pixels of the image which will be imperceptible to the human eye. One way to achieve this goal is to introduce the message by changing the Least Significant Bits (LSB) of each pixel sequentially or randomly [5], [6]. Currently, steganography is done adaptively, that is, it takes into account the content of the image to introduce the message in regions where it is more difficult to be detected by the steganographers. The most employed algorithms in this domain are HUGO [7], HILL [8], MiPOD [9], S-UNIWARD [10] and WOW [11]. **Figure 1** shows a stego image compared to a cover image after the steganographic process, using the S-UNIWARD algorithm with a payload (number of embedded changes) of 0.4 bits per pixel (bpp). On the right side of the figure, the difference in images is shown to illustrate the effect of the algorithm on the stego image.

There are significantly used transformations from the frequency domain (JPEG - Joint Photographic Experts Group) to make steganography, such as Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT) and Singular Value Decomposition (SVD), all explained in [13]. JPEG is the most common loss compression format for images produced by digital cameras, scanners, and other photographic capture devices, which is based on DCT. Some coefficients of the used transform are changed to insert messages in the JPEG domain in such a way that it is imperceptible to the human eye. The most employed algorithms in this domain are J-UNIWARD [10], F5 [14], UED [15] and UERD [16].

Steganalysis consists of detecting whether or not an image has a hidden message. In [13], there is a more in-depth explanation of steganography and steganalysis with their respective classifications. Steganography is traditionally divided into two stages. Stage one consists on the manual extraction of features where the best results have been achieved using Rich Models (RM) [17]. Stage two is based on a binary classifier (an image is steganographic or not) where Ensemble Classifiers (EC) [18], Support Vector Machines (SVM) [19] or perceptrons [20] are typically used. Thanks to advances in Deep Learning (DL) [21] and Graphic Processing Units (GPUs) [22], researchers have begun to apply these techniques in steganography and steganalysis obtaining better detection percentages of steganographic images. When DL is employed in steganalysis, the feature extraction stage and classification are unified under the same architecture, and the parameters are optimized simultaneously, allowing the complexity and dimensionality introduced by manual feature extraction to be reduced [17]. **Figure 2** shows the general structure of steganalysis with manual extraction of characteristics (top side) and steganalysis unifying extraction and classification under the same architecture (bottom side).

A. BACKGROUND

The first application of DL to steganalysis was developed in 2014 by *Tan and Li* [23] whose approach used unsupervised learning from a stack of Auto-Encoders training a Convolutional Neural Network (CNN). Supervised learning

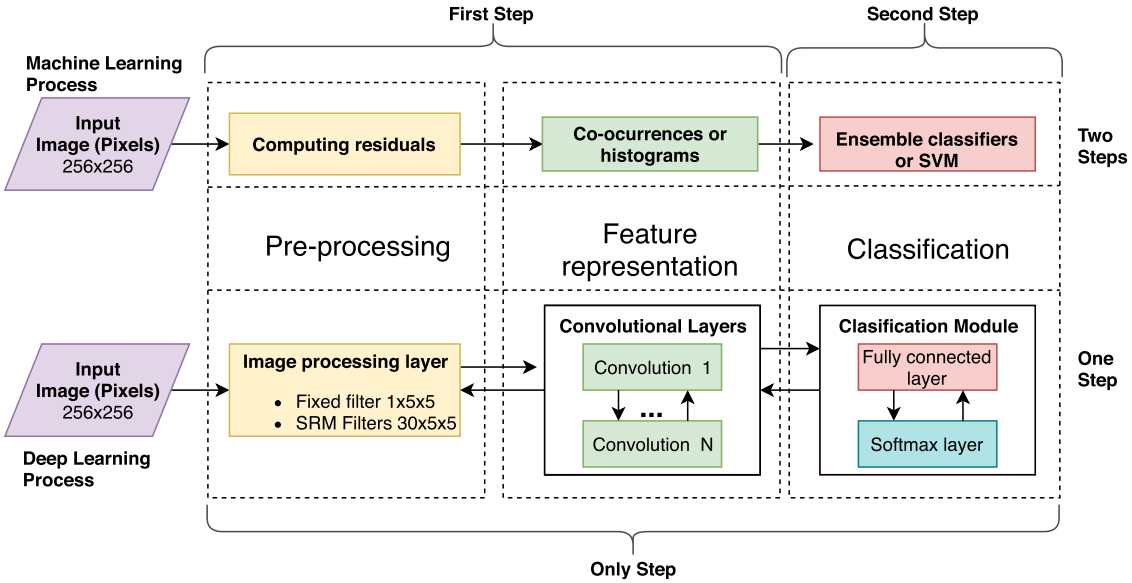

FIGURE 2. Steganalysis based on manual extraction of characteristics (top side) and steganalysis based on deep learning techniques (bottom side).

was then used by pre-processing the image using a High Pass Filter (HPF) to increase the steganographic noise power introduced by the embedding process. The detection percentages of steganographic images were approx. 17% lower than those obtained by Spatial Rich Models (SRM) [17], and approx. 11% higher than those obtained by Subtractive Pixel Adjacency Matrix (SPAM) [24].

In the 2015 *Qian, Tan et al.* [25] designed the first CNN with a supervised learning approach, which consisted of 5 convolutional layers and a specific activation function known as *Gaussian Activation*. The detection percentages of steganographic images were approx. 4% lower than those obtained by SRM [17], and approx. 10% higher than those obtained by SPAM [24].

Then in 2016 *Pibre, Chaumont et al.* [26] took over *Qian's* work and proposed two new neural networks. The first one was a 2-layer CNN and the second a Fully Connected Neural Network (FNN) composed of 2 layers. Their experiments were characterized by using the same encryption key. *Xu, Shi et al.* [27] proposed a CNN similar to *Qian's* with 5 convolutional layers. Unlike that network, *Xu, Shi et al.* used an absolute value layer (ABS) and 1×1 convolutional kernels to strengthen the statistical modeling and obtain better results. *Xu, Shi et al.* took their proposed network and used it as a Base Learner [28] to train sets of CNNs in order to obtain better training parameters and further improve their detection results. In the same year, *Qian, Tan, et al.* used Transfer Learning [29] exchanging the parameters of a CNN, which was trained with steganographic images with a high payload, to another CNN that would be trained to detect images with a low payload. The results obtained improved compared to CNNs that did not use Transfer Learning, but still would not surpass the traditional algorithms. All the advances obtained previously were implemented in the spatial domain. After that, the researchers have focused on doing steganalysis using DL techniques in the frequency domain (JPEG).

In 2017 *Zeng, Huang et al.* [30], [31] proposed a CNN approach to perform steganalysis in JPEG format images using an RM-inspired pre-processing applied to large sets of images offered by ImageNet [32]. The results obtained were close to those recorded in the literature. In the same year *Chen, Fridrich, et al,* built a new network using Phase-Split inspired by the JPEG compression process [30]. A CNN assembler was used to obtain results significantly higher than those obtained by state of the art. Subsequently *Xu* [33] proposed a new CNN inspired by ResNet [34] consisting of 20 convolutional layers followed by a Batch Normalization (BN) process [35], [36]. *Tang, Li et al.* [37] suggested to make steganography of images in the spatial domain taking as reference two networks that compete with each other. This methodology, known as Generative Adversarial Network (GAN), used the rivalry between steganography and steganalysis (2 competing networks) to automatically learn which was the best position in where to embed a message. *Ye, Yi et al* [38] proposed a new CNN in the spatial domain with 8 convolutional layers, a self-activation function known as Truncation Linear Unit (TLU), and filter banks for image pre-processing. These filter banks initialize their SRM-based weights to obtain residual characteristic maps and avoid the use of the static filter used by all previous CNNs. The trend in 2017 was to train sets of CNNs and modify the network architecture to mimic the SRM feature extraction process. Another significant contribution was to jump between different convolutional layers (ResNet [34], [39]) thus enabling deeper CNNs to be designed, ensuring network convergence and improving detection accuracy; until then, detection results

were improved by approximately 10% compared to those recorded in the literature.

In 2018 a new CNN was proposed in the spatial domain by *Yedroudj, Chaumont et al.* [40]. This CNN brings together the best features of its predecessors (a set of input filters for pre-processing based on SRM feature extraction, 5 convolutional layers, BN, TLU activation units and an increase in the size of the training database) to get better results from those reported by literature. In [41] *Tsang, Fridrich, et al.* take *Ye's* network and modify it to be able to classify high-resolution steganographic images from CNNs training with low resolution images. *Yedroudj, Chaumont et al.* [42] studied the effect of enriching the database traditionally used in steganalysis known as *BOSSBase* [43]. The added images belong to the *BOWS2* [44] database, as well as images captured with cameras with similar characteristics to those used to create the traditional database. Finally, the number of images in both databases were increased using cropping, resizing, rotation and interpolation operations. They concluded that to improve the performance of steganalysis it is recommended to have a large database acquired with similar cameras and dimensions. *Chen, Fridrich, et al.* [45] propose to do quantitative steganalysis using DL techniques to predict the payload contained in a steganographic image in both spatial and frequency domain (JPEG). *Li et al.* [46] propose to combine 3 CNNs in parallel. Each network uses a different pre-processing layer for feature extraction (Gabo Filters [47], Linear-SRM [17], nonlinear-SRM [17]) and simultaneously uses 3 activation functions (ReLU [48], Sigmoid [49] and TanH [49]) in order to consider more pre-processed information. *Zeng et al.* [50] make an experiment similar to the previous one on color images. *Boroumand Fredrich et al.* [34] proposed a new CNN that avoids as much as possible the use of tricks, such as using SRM filters for pre-processing. This network works in both the spatial and frequency domains. *Zhang et al.* [51] suggested a new CNN that optimizes the weights of the pre-processing layer filters to increase the power of steganographic noise and decrease image content. It uses separate convolutions to obtain residue channel correlations and spatial correlations separately for better feature representation, and finally uses Spatial Pyramid Pooling (SPP) [52] to add local features, to improve feature representation capability, and to allow arbitrary image sizes.

The rest of the paper has the following order: **Section II** explains how the systematic review of the literature was done. **Section III** presents the results found and the state of the art. **Section IV** shows some conclusions and future work.

II. LITERATURE SYSTEMATIC REVIEW

This review is aimed at showing how the application of DL to steganalysis has evolved in the recent years, highlighting the most significant results and possible future work. The reader is recommended to previously scan the review in [53] which explains in detail the basic methods behind steganography and steganalysis. This literature review used the phases proposed in [54] which are described below.

A. A NEED FOR BIBLIOGRAPHIC REVIEW

Due to technological advances in hardware and software applied to Artificial Intelligence (AI), the constant development of the GPUs [22], and the emergence of frameworks supporting the design and implementation workflow of machine learning algorithms, such as TensorFlow [55], researchers have started to use these techniques and technologies thoroughly. DL applied to steganalysis is no exception and, since 2014, researchers in this area have been using them in the detection of steganographic images. Consequently, a great variety of experiments have been implemented, obtaining percentages of detection that have overcome up to 13% to those obtained by SRM, increasing the overall interest in this topic.

B. RESEARCH QUESTIONS

In order to identify the current status of DL applied in steganalysis, it is key to obtain data on scientific production, journal articles or conference proceedings in a systematic manner as it is the purpose of this review. Relevant articles were searched in different databases for this purpose, and the following research questions arose:

Q1: Is the use of CNNs for steganographic image detection beneficial over traditional methods in terms of detection performance?

Q2: Which are the different architectures and novel components of CNNs used to make steganalysis of digital images?

Q3: What are the detection percentages of the steganographic image using CNNs?

Q4: What are the most used digital image databases to do DL experiments on steganalysis?

C. BIBLIOGRAPHIC SEARCH

A systematic review of documents from scientific societies devoted to forensic science, computer security, digital signal processing, digital image processing, and AI was conducted. The search for information took into account authors who were researchers, students or professors, with the aim of obtaining a list of articles that explain the design and implementation of CNNs to do steganalysis.

The terms chosen for this search were:

- **Deep Learning**
- **Convolutional Neural Network**
- **Steganalysis**

With the search terms defined above, query strings, which are complemented with logical operators, were built to improve execution results. The search process was limited to articles published in journals or conference proceedings between $2014 - 2018$, only in the English language.

The general search string is listed below

((“Deep Learning” OR “Convolutional Neural Network”) AND (“Steganalysis”))

In **Table 1** the databases and search strings used for the review are shown. The search for grey literature included all types of documents contributed by the most relevant researchers in the area. In the presentations of

TABLE 1. Databases and search strings for literature review.

Name of the search query or database	Search string
IEEExplorer	(((deep learning) OR convolutional neural network) AND steganalysis)
Cornell University Library	((TitleCombined:(deep learning)) OR (TitleCombined:(convolutional neural network))) AND ((TitleCombined:(steganalysis)) OR (Abstract:(deep learning)) OR (Abstract:(convolutional neural network))) AND (Abstract:(steganalysis))
The ACM Digital Library	acmdlTitle:(+deep +learning +steganalysis) OR acmdlTitle:(+convolutional + neural +network +steganalysis) OR recordAbstract:(+deep + learning +steganalysis) OR recordAbstract:(+convolutional + neural +network +steganalysis) OR content.ftsec:(+deep +learning + steganalysis) OR content.ftsec:(+convolutional + neural +network +steganalysis)
Google Scholar	allintitle: deep learning steganalysis
ScienceDirect	pub-date >2014 and TITLE-ABSTR-KEY ((deep learning OR convolutional neural network)) and TITLE-ABSTRKEY((steganalysis))
Springer	'"deep learning" OR "convolutional neural network" AND steganalysis'
Scopus	TITLE-ABS-KEY (deep AND learning AND steganalysis)
Web Of Science	(TEMA: (deep learning steganalysis) OR TITULO: (deep learning steganalysis))

congresses, symposia or short courses exposed in [42], [56]–[60], [80], a good baseline was found to start with a chronological review of the literature.

D. INCLUSION AND EXCLUSION CRITERIA

The inclusion criterion was taken into account:

- Articles written in English
- Articles published between 2014 – 2018
- Articles published as results of Conferences, Congresses or Journals.
- Articles only included in the databases in **Table 1**.
- Articles which use DL applied to steganalysis.

It was taken into account as an exclusion criterion:

- Articles which only have a table of contents and summary.
- Articles not related to research.
- Articles where steganalysis is done without applying DL.

E. EXTRACTION AND EVALUATION OF INFORMATION

After a systematic search of the literature on information sources and with the search strings provided in **Table 1**, 312 items were found distributed as shown in **Table 2** and their percentage distribution is shown in **Figure 3**.

The 79 articles obtained by the search on the *IEEExplore* database were taken as a reference, classified by year and

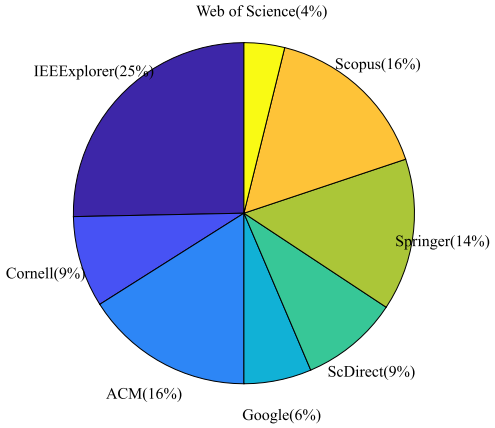

FIGURE 3. Percentage of articles found with the search strings used in the different databases.

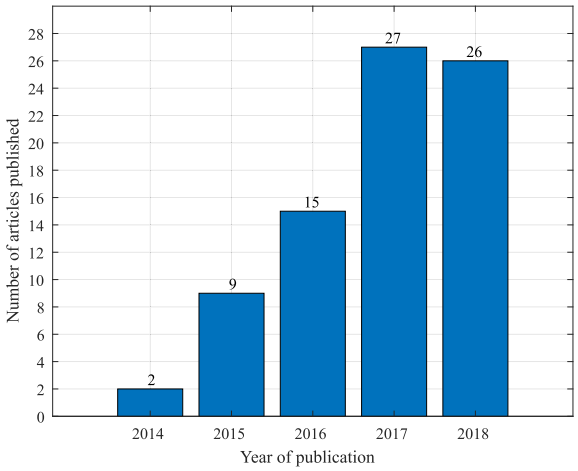

FIGURE 4. Number of articles published in IEEExplore with respect to the year.

their results are shown in **Figure 4** which demonstrated a trend of publication of articles in the area of interest.

It is important to clarify that a large number of repeated articles were found in different search sources. After unifying the repeated articles, a base of 110 articles was obtained.

The steps for systematic literature search are listed below:

1) **Start**: Defining search strings and selecting databases.
2) **Search**: Execution of search strings in the selected databases (312 articles were found).
3) **Repeated**: Unify the articles that appear in more than one database (the results are condensed to 110 articles).
4) **Title**: After reading the title, 24 are accepted and 86 articles are rejected.
5) **Abstract**: After reading the abstract, 18 are accepted and 6 articles are rejected.
6) **Full text**: After reading the entire text, 14 articles are accepted and 4 articles are rejected.

The 14 articles chosen for the systematic review of the literature are highly reliable as they are found in high-impact international databases, they also have a good number of citations and their authors have great prestige in the subject.

TABLE 2. Systematic literature search results.

Name of the search query or database	Filters applied for the search	Number of publications found
IEEExplore	All text and Metadata	79
Cornell University Library	Title and abstract	27
The ACM Digital Library	All text	50
Google Scholar	Title	20
ScienceDirect	All text	29
Springer	All text	45
Scopus	Title, abstract and keywords	50
Web Of Scienc	Title and subject	12
Total		**312**

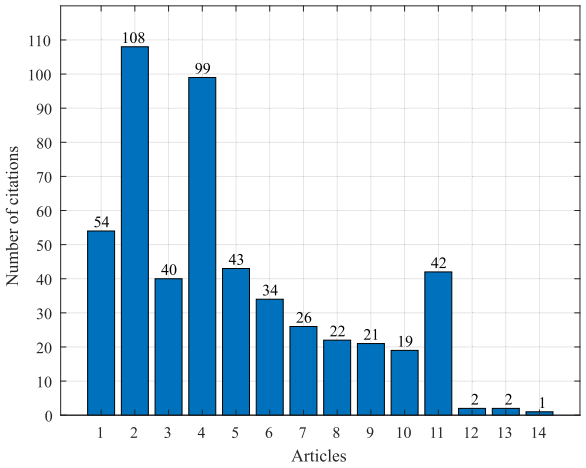

FIGURE 5. Number of citations for each of the items selected for literature review.

In **Table 3**. the title of the article, the authors and the year of publication are shown chronologically. For the rest of the document, when one of these articles is needed, the list given in this table will be used as a convention. The articles were organized through the *Mendeley* application.

Finally, the selected articles were taken, and the number of citations was extracted according to *Google Scholar*, the results are shown in **Figure 5**.

III. DEVELOPMENT OF THE SUBJECT

After a systematic search of the literature using the criteria of **Section II**, the development of the topic was focused in 14 articles which have a common thread in the advances that the DL applied to steganalysis has had. The place of publication and main contributions are shown in **Table 5**. It can be observed that the topic has become relevant due to the frequency of publications in the last 2 years. The researchers who have contributed most to this topic are as follows: *Jessica Fridrich, Marc Chaumont, Yinlong Qian, Guanshuo Xu, Tieniu Tan, Shunquan Tan, Yun-Qing Shi, Jishen Zeng, Mo Chen, Bin Li, Jiwu Huang, Jian Ye, Jiangqun Ni*. The results are published in high-impact symposia, congresses, and journals. The main contributions are the generation of

TABLE 3. Articles found by systematic literature search.

No Art.	Title of the article	Authors	Date of publication
1	Stacked Convolutional Auto-Encoders for Steganalysis of Digital Images [23]	Shunquan Tan, Bin Li	12/12/2014
2	Deep learning for Steganalysis via Convolutional Neural Networks [25]	Yinlong Qiana, Jing Dong, Wei Wang, and Tieniu Tan	4/03/2015
3	Deep Learning is a Good Steganalysis Tool when Embedding key is Reused for Different Images, even if there is a Cover Source-Mismatch [26]	Lionel Pibre, Jérome Pasquet, Dino Ienco, Marc Chaumont	18/02/2016
4	Structural Design of Convolutional Neural Networks for Steganalysis [27]	Guanshuo Xu, Han-Zhou Wu, Yun-Qing Shi	30/03/2016
5	Ensemble of CNNs for Steganalysis: An Empirical Study [28]	Guanshuo Xu, Han-Zhou Wu, Yun-Qing Shi	30/06/2016
6	Learning and Transfering Representations for Image Steganalysis Using Convolutional Neural Network [29]	Yinlong Qiana, Jing Dong, Wei Wang, and Tieniu Tan	28/09/2016
7	Large-Scale JPEG Image Steganalysis Using Hybrid Deep-Learning Framework [30]	Jishen Zeng, Shunquan Tan, Bin Li, Jiwu Huang	26/01/2017
8	JPEG-Phase-Aware Convolutional Neural Network for Steganalysis of JPEG Images [61]	Mo Chen, Vahid Sedighi, Mehdi Boroumand, Jessica Fridrich	22/06/2017
9	Deep Convolutional Neural Network to Detect J-UNIWARD [33]	Guanshuo Xu	22/06/2017
10	Automatic Steganographic Distortion Learning Using a Generative Adversarial Network [37]	Weixuan Tang, Shunquan Tan, Bin Li, Jiwu Huang	10/10/2017
11	Deep Learning Hierarchical Representations for Image Steganalysis [38]	Jian Ye, Jiangqun Ni, Yang Yi	1/11/2017
12	Yedroudj-Net: An Efficient CNN for Spatial Steganalysis [40]	Mehdi Yedroudj, Frédéric Comby, Marc Chaumont	20/04/2018
13	Steganalyzing Images of Arbitrary Size with CNNs [41]	Clement Fuji, Jessica Fridrich	20/04/2018
14	Efficient feature learning and multi-size image Steganalysis based on CNN [51]	Ru Zhang, Feng Zhu, Jianyi Liu and Gongshen Liu	30/07/2018

TABLE 4. Error percentage of the CNNs and SRM for two steganographic algorithms with a payloads of 0.4bpp and 0.2 bpp [25], [27], [38], [40], [51].

Network	WOW 0.2 bpp	WOW 0.4 bpp	S-UNIWARD 0.2 bpp	S-UNIWARD 0.4 bpp
SRM+EC (2012)	36.5	25.5	36.6	24.7
QianNet (2015)	38.6	29.3	46.3	30.9
XuNet (2016)	32.4	20.7	39.1	27.2
YeNet (2017)	33.1	23.2	40.0	31.2
YedroudjNet (2018)	27.8	14.1	36.7	22.8
ZhuNet (2018)	23.3	11.8	28.5	15.3

different CNNs, which have evolved thanks to the contributions of the predecessor networks. The CNNs proposed so far in chronological order are **QianNet** or **GNCNN** [25], **XuNet** [27], **YeNet** [38], **Yedroudj-Net** [40], **ZhuNet** [51].

Table 6 shows the architecture implemented by each author, the database used for training, validation and testing, the domain of the experiment (spatial or frequency), the steganographic algorithms used for the steganalysis and the best results.

It can be observed from **Tables 5** and **6** that the first experiment was done using unsupervised learning by implementing an Auto-Encoders stack. Work continued on supervised learning ever since following 3 fundamental principles for steganalysis: reinforcement of the steganographic noise, by means of a fixed high-pass filter, extraction of characteristics and classification; all unified under a single architecture which optimizes its parameters simultaneously. The first advances in the subject were made in the spatial domain, and, then, the researchers entered the frequency domain (JPEG).

Researchers have tested different ideas using CNNs for their experiments, among the most important of which are as follow: increasing the height of the network or using fully connected networks [26]; using custom activation features to ensure network convergence and improve steganographic image detection rates [25], [38], [40]; using CNNs with jumps between convolutional layers (Residual Networks or Dense Networks) in order to design very deep networks (20 or more layers) achieving network convergence and improving detection percentages [33], [34], [62]–[66]; training sets of CNNs and transferring the learned parameters to CNNs where their convergence is complex or they have a low detection percentage [28], [29], [61]; training CNNs with a certain database and test the network with a completely different database in order to determine the reliability of the designed CNNs (Cover-Source Mismatch) [26], [61]; strengthening statistical modeling by means of an absolute value layer (ABS) [27], [28], [40], [51]; improving steganographic noise extraction by using filters designed in SRM and doing feature extraction and classification with CNNs [30], [38], [40], [51]; using real-world databases such as ImageNet to see how well a CNN can adapt to any dataset with diverse resolution and capture characteristics [25], [30], [33], [34], [62], [63]; placing two CNNs to compete. In this case the first network is used for steganography and the second for steganalysis, the aim is to obtain an automatic steganography process due to the learning of the characteristics of both processes [36], [37], [67]–[72]; training a network to be able to classify high-resolution images from low-resolution images [41]; predicting the payload (quantitative steganalysis) of a steganographic image using DL in the spatial and JPEG domain [45], [73]; generating an increase in the database taking into account trimming, rotation and interpolation operations, as well as the use of cameras with similar or different characteristics for image acquisition, taking care with resizing [30], [38], [51], [74]; placing 3 CNNs to work in parallel [46], each network uses the activation functions (ReLU, Sigmoid and TanH) and different filters in the pre-processing layer inspired by Gabo Filters [47] and SRM (linear and non-lienar) [17]; doing a similar work to the previous one but in color images [50], among others.

Most of the proposed networks use the high-pass filter of **Equation 1**, developed in [17] and used for steganalysis for the first time [25]. High-Pass filter parameters are not optimized during the training process. This filter pre-processes the image to strengthen the steganographic noise and decrease the impact of the image content. This filter helps the CNN training to be convergent; in case the convergence is not used it can be slower or non-existent. The last designed CNNs do not use this filter; they use a bank of filters proposed by the SRM to obtain maps of residual characteristics instead.

$$K = \frac{1}{12}\begin{pmatrix} -1 & 2 & -2 & 2 & -1 \\ 2 & -6 & 8 & -6 & 2 \\ -2 & 8 & -12 & 8 & -2 \\ 2 & -6 & 8 & -6 & 2 \\ -1 & 2 & -2 & 2 & -1 \end{pmatrix} \tag{1}$$

The general-level operation done on a CNN can be seen in **Equation 2**. Where M^l is one of the maps of characteristics of the l-th layer, ${M_i}^{l-1}$ is the i-th map of characteristics of the previous layer, ${K_i}^l$ is the i-th kernel of the l-th layer, b^l is the bias parameter of the l layer, $*$ is the convolution operation, $f()$ is the non-linear operation known as the activation function, $pool()$ is the pooling operation, and $norm()$ is the normalization operation. The order of the operations in the convolutional layers are convolution, normalization, activation function, and pooling. Feature maps obtained by the last layer are passed to the classification module which consists of one or several layers of neurons completely connected and a Softmax layer; the last layer is in charge of normalizing the values of the CNN between [0, 1], which in turn are the ones of probability that indicate if the image is cover or stego.

$$M^l = norm\left(pool\left(f\left(\sum_{i=1}^{n}\left(M_i^{l-1} * K_i^l\right) + b^l\right)\right)\right) \tag{2}$$

The nonlinear activation functions employed in the CNNs studied are Rectification Linear Unit (*ReLU*) [48], Tangent Hyperbolic (*TanH*) [49], Gaussian and TLU. This last activation function is exclusive of DL applied to steganalysis, and its function is to limit the range of values and avoid the modeling of the network to large values. Normally *TanH* is used in the first layers and *ReLU* in the last ones.

The operation used for data normalization is BN, which is summarized in **Equation 3** [35]. BN consists of normalizing the distribution of each feature of a feature map so that the average is zero and the variance is unitary, and possibly, if necessary, allows re-scaling and re-translation of the distribution.

Given a random variable X whose realization is a value $x \in \mathbb{R}$ of the feature map, the BN of this value x is:

$$BN(x, \gamma, \beta) = \beta + \gamma \frac{x - E[X]}{\sqrt{Var[X] + \varepsilon}} \tag{3}$$

with $E[X]$ the expectation, $Var[X]$ the variance, and γ and β two scalars representing a re-scaling and an re-translation.

TABLE 5. Contributions of the main articles that apply DL to steganalysis.

No Art	Journal or Conference	Main Contributions
1	Signal and Information Processing Association Annual Summit and Conference (APSIPA 2014), Asia-Pacific	The first approach of a CNN applied to steganalysis using a stack of convolutional Auto-Encoders for pre-training. In this paper, it is explained that the traditional methods to do steganalysis such as SRM [17] are similar to the structure of a CNN. It still does not reach the results offered by SRM [17], but it surpasses the results offered by SPAM [24], the two best steganalyzers of the moment with manual extraction of characteristics.
2	SPIE/IS&T Electronic Imaging (EI 2105), Media Watermarking, Security, and Forensics.	The first CNN with supervised learning is proposed. It uses a high-pass filter to reinforce steganographic noise and decrease image content. For the extraction of features, there are 5 convolutional layers, a custom Gaussian activation function, and Average Pooling. For classification, a module of neurons fully connected to a Softmax layer was added. The results are competitive to state of the art (SRM and SPAM). The name of this network is **QianNet** or **GNCNN**.
3	Media Watermarking, Security, and Forensics, IS&T Int. Symp. on Electronic Imaging (EI 2016)	Returns the **QianNet** network [25] as a basis for experimentation and after testing 40 designs of different Neural Networks, proposes two new networks to do steganalysis, which, trained under the scenario Clairvoyant [26] (for example using the same incrustation key to do the steganography process) or under the scenario Cover-Source Mismatch [26] (training with a database and testing with a completely different one) are achieved better results than those obtained by SRM. The proposed networks are characterized by their higher, shallower depths, the Gaussian activation function is changed to the classic ReLU [48] and the Pooling step [75] is suppressed. The best proposed CNN consists of two convolutional layers, the first convolutional layer applies 64 kernels 7x7 that work as a band-pass filter, the second convolutional layer applies 16 kernels 5x5 to obtain insensitive features for the Cover-Source Mismatch effect, this task subdivision cannot be achieved with traditional methods generating an inferior performance over other data sets. Finally, CNNs can use transfer learning to test other data sets, and this is not possible with traditional methods.
4	IEEE Signal Processing Letters (2016)	A new CNN called **XuNet** [27] is proposed. This network is characterized by the fact that after the first convolutional layer it uses an ABS layer ABS to facilitate and improve the statistical modeling taking into account the sign symmetry [17] existing in the noise residuals. Additionally, it uses BN to prevent CNN training from falling to poor local minima, and to learn optimal scales and biases for feature maps [35]. A TanH [49] activation function is also used on the first two layers, and a ReLU [48] activation function on the rest of the layers, in order to reinforce statistical modeling and avoid overfitting. These activation functions are also used to avoid low slope regions and the cancellation of the gradient value when using back-propagation (gradient vanishing phenomenon), which makes learning impossible. Finally, convolutions 1x1 are used in the last layers in order to limit statistical modeling.
5	IH&MMSec Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security (2016)	It is proposed to train a set of CNNs that learn about common characteristics (characteristics vector), output probabilities and information lost by the pooling operation to obtain a more precise classification. The results obtained when training a set of CNNs provide better results than when training a single model. The set of trained networks uses a structure very similar to the one proposed in [27], with the difference of adding a layer of convolutions and increasing the size of the pooling of the last two layers.

TABLE 5. *(Continued.)* **Contributions of the main articles that apply DL to steganalysis.**

No Art	Journal or Conference	Main Contributions
6	IEEE International Conference on Image Processing (ICIP 2016)	The parameters learned in the convolutional layers, and in the fully connected layers of a high payload, CNN for a given steganographic algorithm is transferred to train a low payload CNN for the same algorithm, thus improving the performance of this type of networks.
7	IEEE Transactions on Information Forensics and Security(2017)	It is proposed for the first time to do steganalysis using DL in the frequency domain (JPEG). A hybrid Framework is generated where in the first stage a manual extraction of characteristics is made from a bank of filters supplied by SRM. Specifically, it corresponds to the phase of convolution, quantization, and truncation proposed by DCTR in [78] for RM. For the second stage, a classification is made using 3 convolutional subnets, followed by 3 completely connected network layers and a Softmax layer. The experimentation is done on large-scale databases (ImageNet) to obtain results closer to the real world, trained with up to 5 million images. The training was done with 5 versions of DL models independently to combine the results then and obtain better accuracy (Ensemble of CNNs). Finally, the learned model can be easily transferred to a different attacking target and even to a different data set obtaining satisfactory results.
8	IH&MMSec Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security (2017)	Knowledge of JPEG phases is incorporated into the architecture of a CNN to increase accuracy. The **XuNet** network is taken and adapted to work in the JPEG domain, at the end two CNNs (**PNet**, **VNet**) are proposed. The first allows each JPEG phase to pass through a CNN thus increasing the computational complexity; the second allows to mix the JPEG phases and thus decrease the computational complexity. Both networks will be trained individually and using sets of CNNs to obtain better results. Another innovative concept introduced is the "Catalyst Kernel" which, together with the traditional high pass filters used to pre-process images, allows the network to learn the essential kernels for the detection of the stego signal introduced by JPEG steganography. Experiments with J-UNIWARD and UED-JC inlay algorithms are used, and the results are compared with the traditional steganalysis method Selection-Channel-Aware Gabor Filter Residuals (SCA-GFR) [79]. For network training, parameters were transferred (Transfer Learning) from the network training with 0.4 bpnzac and with these parameters already trained, initialize the other networks. Finally, we would like to know what effect it has on CNN to train with an image database and try a completely different one (Cover Source-Mismatch) [26], [20].
9	IH&MMSec Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security (2017)	A CNN with 20 convolutional layers is proposed. It is demonstrated that deep CNNs and Pooling operation can overcome traditional methods based on manual feature extraction. This network is tested on J-UNIWARD. It is also proposed to Res-Net in order to avoid the disappearance of the gradient due to the depth of the net. The experiments are performed on ImageNet's CLS-LOC base with approximately 1 million 256X256 cropped images compressed with a QF 75 quality factor.
10	IEEE Signal Processing Letters (LSP 2017)	It is proposed to do automatic steganography taking into account the characteristics of adaptative steganography. Two CNNs are proposed to compete with each other, the first is used for steganography (Generator), and the second for steganalysis (Discriminator), through this competition the algorithm can automatically embed a message in locations where it is more difficult to make the detection for a steganalyzer. Through the other training of these two opposing subnets, the proposed framework can automatically learn to embed change probabilities for each pixel in a given cover image in the spatial domain. Automatic Steganographic Distortion Learning framework with GAN (ASDL-GAN) simulates the rivalry between additive distorted steganography and DL steganalysis.

TABLE 5. *(Continued.)* **Contributions of the main articles that apply DL to steganalysis.**

No Art	Journal or Conference	Main Contributions
11	IEEE Transactions on Information Forensics and Security (TIFS 2017)	A CNN is proposed that does not use the traditional high-pass filter to obtain the steganographic noise, on the contrary, it uses a set of high-pass filters used in the calculation of residual maps of SRM, whose values are used to initialize the trainable filters instead of doing it randomly. The purpose of these filters is to suppress image content and amplify steganographic noise effectively. A new activation function called TLU [38] is adopted in order to increase the signal-to-noise ratio (SNR) [88] which is extremely low in the steganography incrustation process. Finally, the performance of the CNN-based steganalyzer is increased by incorporating knowledge of channel selection (knowledge of the probability of change of each pixel) [61] and parameter transfer for low payload networks. By adding the values of the probabilities of change of each pixel and the characteristic maps generated by the pre-processing filters (filters initialized with SRM values), it is possible to deliver more information about steganographic noise to the following convolutional layers and thus improve the performance of CNN. This network is called **YeNet**.
12	International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2018)	This network unites the best features between the **XuNet** and **YeNet**. The use of a filter bank for pre-processing based on SRM, TLU and BN activation function is highlighted. It does not use the knowledge of the channel selection (map of probabilities of change of a pixel). For the training, they use an extended database to improve the results. This network is called **Yedroudj-Net**.
13	Media Watermarking, Security, and Forensics (2018)	The **YeNet** network is taken and adapted to train with sets of small resolution images and generalize the model to be able to do steganalysis in high-resolution images, that is to say, it addresses the problem that the images have different sizes, this due to the tremendous computational cost involved in training a CNN with high-resolution input images.
14	Computing Research Repository (2018)	A new CNN is proposed that for the first time optimizes the kernel weights of the pre-processing layer to increase the steganographic noise signal. Convolutional layer filters are reduced in size to decrease the number of parameters and model characteristics in a small local region. Separable convolutions [89], [90] are used to residue channel correlation, spatial correlation, compress image content and increase signal-to-noise ratio. SPP [52] is used to add local features, improve feature rendering capability, and allow arbitrary image sizes. Finally, the database is increased to improve detection accuracy. This network is called **ZhuNet**. The results obtained by this network exceed those obtained by **XuNet**, **YeNet**, **YedroudjNet** and **SRM+EC**.

The expectation $E[X]$ and the variance $Var[X]$ are updated at each batch, while γ and β are learned by back-propagation. In practice, the BN makes the learning less sensitive to the initialization of parameters [35], allows to use a higher learning rate which speeds up the learning speed, and improves the accuracy of the classification [61]. In the first proposed CNNs, BN is not used.

Average Pooling [75] operation is commonly used in all CNNs for pooling due to the steganographic noise introduced by the incrustation process is very weak and using this operation favors the propagation and preservation of this type of noise, which does not occur in the case of using Max Pooling [75]. The pooling customarily used is a local operation that is computed with its neighbors.

The most important CNNs obtained from the studies are as follows: **QianNet** or **GNCNN** (2015) [25], [76], **XuNet** (2016) [27], **YeNet** (2017) [38], **YedroudjNet** (2018) [40] and **ZhuNet** (2018) [51], all were initially designed in the spatial domain and some were adapted to work in the frequency domain (JPEG). **QianNet** is characterized by having 5 convolutional layers, a Gaussian activation function and Average Pooling after each convolutional layer, 2 fully connected layers, and 1 Softmax. **XuNet** is characterized by having 5 convolutional layers, an ABS layer after the first convolutional layer, using TanH activation functions for the first 2 layers and ReLU for the last 3 layers, BN in each convolutional layer, 2 fully connected layers, and 1 Softmax. **YeNet** uses an SRM filter bank to do the steganographic

TABLE 6. Characteristics of CNNs, databases, steganographic algorithms and results of the main DL articles applied to steganalysis.

No Art	Network architecture	Databases	Domain	Steganographic Algorithms	Percentage of error (Best obtained)
1	• 9 convolutional layers subdivided into 3 stages. • Max Pooling. • 1 layer completely connected. • 1 Softmax.	BOSSBase	Spatial	HUGO	Using 0.4 bpp on BOSSBase **CNN**=31 **SPAM**=42 **SRM**=14
2	• 1 pre-processing layer (1 high-pass filter). • 5 convolutional layers. • Gaussian activation function. • Average Pooling. • 3 completely connected layers. • 1 Softmax.	BOSSBase ImageNet	Spatial	HUGO WOW S-UNIWARD	Using 0.4 bpp on BOSSBase **CNN**=HUGO (28.29),WOW (29.3), S-UNIWARD (30.29) **SRM**=HUGO (25.2), WOW (25.7), S-UNIWARD (26.3) **SPAM**=HUGO (39.1), WOW (38.2), S-UNIWARD (35.1) Using 0.4 bpp on ImageNet **CNN**=HUGO (33.6),WOW (34.1), S-UNIWARD (34.7) **SRM**=HUGO (32.5), WOW (34.7), S-UNIWARD (34.4) **SPAM**=no results
3	**CNN**: • 1 pre-processing layer (1 high-pass filter). • 2 convolutional layers. • ReLU activation function • Without Pooling. • 2 completely connected layers. • 1 Softmax. **FNN**: • 1 pre-processing layer (1 high-pass filter). • 2 completely connected layers. • ReLU activation function. • 1 Softmax.	BOSSBase LIRMMBase	Spatial	S-UNIWARD	Using 0.4 bpp on BOSSBase Clairvoyant scenario **CNN**=7.4 **FNN**=8.66 **SRM**=24.67 Using 0.4 bpp on BOSSBase (Train) and LIRMMBase (Test) Cover-Source Mismatch scenario **CNN**=5.16 **FNN**=5.89 **SRM**=48.29
4	• 1 pre-processing layer (1 high-pass filter). • 5 convolutional layers. • ABS Absolute Value Layer (Only after the first convolutional layer). • Activation function TanH, ReLU. • Batch Normalization BN. • Average Pooling. • 2 completely connected layers. • 1 Softmax.	BOSSBase	Spatial	S-UNIWARD HILL	Using 0.1 bpp on BOSSBase **CNN**=S-UNIWARD (42,67), HILL (41.56) **SRM**=S-UNIWARD (40.75), HILL (43.56) Using 0.4 bpp on BOSSBase **CNN**=S-UNIWARD (19,76), HILL (20.76) **SRM**= S-UNIWARD (20.47), HILL (24.53)
5	• 1 pre-processing layer (1 high-pass filter). • 6 convolutional layers. • ABS absolute value layer (Only after the first convolutional layer). • Activation function TanH, ReLU. • Batch Normalization BN. • Average Pooling. • 1 layer completely connected. • 1 Softmax	BOSSBase	Spatial	S-UNIWARD	Using 0.4 bpp on BOSSBase **CNN** = 18.99 **SRM**=18.97

TABLE 6. *(Continued.)* **Characteristics of CNNs, databases, steganographic algorithms and results of the main DL articles applied to steganalysis.**

No Art	Network architecture	Databases	Domain	Steganographic Algorithms	Percentage of error (Best obtained)
6	• 1 pre-processing layer (1 high-pass filter). • 5 convolutional layers. • Gaussian activation function. • Average Pooling. • 2 completely connected layers. • 1 Softmax	BOSSBase	Spatial	WOW S-UNIWARD	Using 0.4 bpp on BOSSBase **CNN**=WOW (21.95), S-UNIWARD (22.05) **SRM**=WOW (20.67), S-UNIWARD (20.55)
7	The proposed network consists of two stages: The first stage is manual feature extraction based on Rich Models (convolution Phase and Quantization & Truncation). The second stage consists of 3 convolutional sub-networks of deep learning for classification each with: • 3 convolutional layers. • ABS absolute value layer (only for after the first convolutional layer). • Batch Normalization BN. • ReLU activation function. • Average Pooling • 3 completely connected layers. • 1 Softmax.	ImageNet	JPEG	J-UNIWARD UERD UED	The article presents the results in graphical form which does not allow to extract the results in a precise way.
8	Two networks are proposed: First Network (PNet), No mixing of channels • 2 pre-processing filters. • 5 convolutional layers, the first 2 layers similar to the XuNet network. • The characteristic maps obtained at the exit of the second convolutional layer are divided into 64 divisions of 16 maps each one which are analyzed by a set of CNN's (layers 3 to 5) in an independent way). • ABS absolute value layer (Only after the first convolutional layer). • Activation function TanH (layers 1 to 2), ReLU (layers 3 to 5). • Average Pooling only for layers 3 to 5. • 1 layer completely connected. • 1 Softmax	BOSSBase BOWS2	JPEG	J-UNIWARD UED-JC	Using 0.1 bpnzAC QF 75 BOSSBase(Train,Test) BOWS2(Test) **CNN-PNet**=J-UNIWARD (35.75), UED-JC (17.77) **CNN-VNet**=J-UNIWARD (36.15), UED-JC (18.97) **SCA GFR**=J-UNIWARD (35.54), UED-JC (22.54) Using 0.3 bpnzAC QF 75 BOSSBase(Train,Test) BOWS2(Test) **CNN-PNet**=J-UNIWARD (12.28), UED-JC (3.90) **CNN-VNet**=J-UNIWARD (13.32), UED-JC (4.07) **SCA GFR**=J-UNIWARD (13.44), UED-JC (6.35) Using 0.4 bpnzAC QF 75 BOSSBase(Train,Test) BOWS2(Test) **CNN-PNet**=J-UNIWARD (6.56), UED-JC (2.34)

TABLE 6. *(Continued.)* **Characteristics of CNNs, databases, steganographic algorithms and results of the main DL articles applied to steganalysis.**

No Art	Network architecture	Databases	Domain	Steganographic Algorithms	Percentage of error (Best obtained)
	Second Network (VNet), With Mixed Channels • 2 pre-processing filters • 5 convolutional layers • ABS absolute value layer (Only after the first convolutional layer). • Activation function TanH (layers 1 to 2), ReLU (layers 3 to 5). • Average Pooling only for layers 3 to 5. • 1 layer completely connected. • 1 Softmax				**CNN-VNet**=J-UNIWARD (7.05), UED-JC (2.32) **SCA GFR**=J-UNIWARD (7.53), UED-JC (3.46) Using 0.5 bpnzAC QF 75 BOSSBase(Train,Test) BOWS2(Test) **CNN-PNet**=J-UNIWARD (3.36), UED-JC (1.33) **CNN-VNet**=J-UNIWARD (3.74), UED-JC (1.20) **SCA GFR**=J-UNIWARD (4.15), UED-JC (1.74)
9	• 16 fixed DCT filters. • ABS absolute value layer. • Activation function: TLU Truncation Linear Unit. • 20 convolutional layers. • BN after each convolution. • ReLU activation function after each convolution. • Global Average Pooling after last convolution. • 1 layer of fully connected neurons • 1 Softmax	BOSSBase ImageNet	JPEG	J-UNIWARD	Using 0.1 bpnzAC QF 75 on BOSSBase **CNN**=32.83 **SCA GFR**=35.98 Using 0.2 bpnzAC QF 75 on BOSSBase **CNN**=19.47 **SCA GFR**=23.16 Using 0.3 bpnzAC QF 75 on BOSSBase **CNN**=11.24 **SCA GFR**=14.09 Using 0.4 bpnzAC QF 75 on BOSSBase **CNN**=6.41 **SCA GFR**=8.07 Using 0.4 bpnzAC QF 75 on ImageNet **CNN**=16.8
10	Structure of the steganalyzer or Discriminator • 1 pre-processing layer (1 high-pass filter). • 5 convolutional layers. • ABS absolute value layer (Only after the first convolutional layer). • Activation function TanH, ReLU. • Batch Normalization BN. • Average Pooling. • 2 completely connected layers. • 1 Softmax Structure of the steganography or Generator	BOSSBase	Spatial	S-UNIWARD ASDL-GAN	Using 0.1 bpp on BOSSBase **CNN**=ASDL-GAN (40.04), S-UNIWARD (42.53) **SRM**=ASDL-GAN (33.02), S-UNIWARD (40.02) Using 0.4 bpp on BOSSBase **CNN**=ASDL-GAN (16.20),

TABLE 6. *(Continued.)* **Characteristics of CNNs, databases, steganographic algorithms and results of the main DL articles applied to steganalysis.**

No Art	Network architecture	Databases	Domain	Steganographic Algorithms	Percentage of error (Best obtained)
	• 1 pre-processing layer (1 high-pass filter). • 25 convolutional layers. • Batch Normalization BN. • Activation function ReLU, Sigmoid. • No Pooling. • 3 completely connected neuron layers.				S-UNIWARD (20.01) **SRM**=ASDL-GAN (17.40), S-UNIWARD (20.22)
11	• In general it has 10 layers • The first layer of 30 filters whose weights are not initialized randomly, but with the values of the high-pass filters used in SRM. The first layer can be merged with a probability map of all pixels in the image to account for channel selection. • 8 convolutional layers. • Activation function ReLU (from layers 2 to 9), TLU (only in the first layer). • Average Pooling from 4 to 7 layers. • 1 layer completely connected. • 1 Softmax.	BOSSBase BOWS2	Spatial	WOW S-UNIWARD HILL	Using 0.1 bpp on BOSSBase+BOWS2(Train and Test) **CNN**=WOW (24.42), S-UNIWARD (32.20), HILL (33.80) **SRM**=WOW (31.63), S-UNIWARD (38.06), HILL (38.94) Using 0.4 bpp on BOSSBase+BOWS2(Train and Test) **CNN**=WOW (9.59), S-UNIWARD (12.81), HILL (17.08) **SRM**=WOW (15.36), S-UNIWARD (21.36), HILL (24.10) Using 0.5 bpp on BOSSBase+BOWS2(Train and Test) **CNN**=WOW (9.06), S-UNIWARD (10.00), HILL (13.05) **SRM**=WOW (13.31), S-UNIWARD (17.32), HILL (21.15)
12	• 30 pre-processing filters based on SRM • 5 convolutional layers • 1 absolute value layer (ABS) only after first layer convolutional • BN after each layer convolutional • Activation function TLU (2 first layers), ReLU (3 last layers) • Average Pooling of layers 2 to 5 • 3 fully connected layers • 1 Softmax	BOSSBase	Spatial	WOW S-UNIWARD	Using 0.2 bpp on BOSSBase **CNN**=WOW (27.80), S-UNIWARD (36,70) **SRM**=WOW (36.50), S-UNIWARD (36.60) Using 0.4 bpp on BOSSBase **CNN**=WOW (14.10), S-UNIWARD (22.80) **SRM**=WOW (25.50), S-UNIWARD (24.70)
13	• 1 layer of 30 filters whose weights are not initialized randomly, but take into account the high-pass filters used in SRM. • 8 convolutional layers. • Activation function ReLU (from layers 2 to 9), TLU (only in the first layer) • 1 layer fully connected • 1 Softmax	BOSSBase	Spatial	LSBM WOW	Using 256x256 on BOSSBase LSBM=11.77 WOW=11.68 Using 512x512 on BOSSBase LSBM=10.68 WOW=13.03 Using 1024x1024 on BOSSBase LSBM=9.40 WOW=14.45

TABLE 6. *(Continued.)* **Characteristics of CNNs, databases, steganographic algorithms and results of the main DL articles applied to steganalysis.**

No Art	Network architecture	Databases	Domain	Steganographic Algorithms	Percentage of error (Best obtained)
14	• 1 layer of 30 filters whose weights are not initialized randomly but takes into account the high-pass filters used in SRM, these filters will be optimized during the training process and decrease the size of the kernels to train fewer parameters. • 2 separate convolution layers to obtain channel correlation and spatial correlation residues, as well as increase steganographic noise power. • 4 convolutional layers. • Batch Normalization of layers 2 to 7 • ReLu activation function of layers 2 to 7. • Average pooling of layers 4 to 6 • 1 Spatial Pyramid Pooling (SPP) module that allows average pooling multi level and work with images of arbitrary size. • 2 fully connected layers. • 1 Softmax	BOSSBase BOWS2	Spatial	WOW S-UNIWARD	Using 0.2 bpp on BOSSBase **CNN**=WOW (23.33), S-UNIWARD (28.50) **SRM**=WOW (36.50), S-UNIWARD (36.60) Using 0.4 bpp on BOSSBase **CNN**=WOW (11.80), S-UNIWARD (15.30) **SRM**=WOW (25.50), S-UNIWARD (24.70) Using 0.2 bpp on BOSSBase+BOWS2(train) BOSSBase(Test) **CNN**=WOW (13.1), S-UNIWARD (17.1) Using 0.4 bpp on BOSSBase+BOWS2(train) BOSSBase(Test) **CNN**=WOW (6.5), S-UNIWARD (8.1)

noise extraction instead of the traditional high-pass filter of **Equation 1**. This CNN consists of 8 convolutional layers, after the first convolutional layer a TLU activation function is used, and for the others, TanH is employed, it has 1 fully connected layer, and 1 Softmax. **YedroudjNet** uses an SRM-inspired filter bank for steganographic noise extraction, 5 convolutional layers, an ABS layer only after the first convolutional layer, TLU activation function in the first 2 layers, ReLU in the last 3 layers, Average Pooling of layers 2 to 5, 2 completely connected layers, and 1 Softmax. This CNN takes the best features of the **XuNet** and **YeNet** and unifies them under the same architecture. **ZhuNet** is characterized by using an SRM-inspired filter bank to initialize the preprocessing layer weights which will be optimized during the training process in order to strengthen the noise introduced by the steganography process and decrease the image content. **ZhuNet** uses separate convolutions to improve the feature extraction process and finally, Average Pooling multi-level known as Spatial Pyramid Pooling (SPP) [52], to allow the network to analyze arbitrary sized images, the results of this CNN outperform the results obtained by **XuNet, YeNet, YedroudjNet** and **SRM+EC**. **Table 4** shows the error percentages of the CNNs mentioned and SRM+EC to detect two algorithms in the spatial domain (S-UNIWARD and WOW) with payloads of 0.4bpp and 0.2bpp. In [34], it is observed a new network design known as **SRNet** which reduces the use of manual devices and heuristics employed by other networks to capture steganographic noise; this network operates in the spatial and frequency domain.

Figure 6 shows the architectures of the most important networks so far. In *purple* it is specified the entry of pixels to CNN. In most experiments the image size is 256 × 256, this due to processing limitations and computational memory. The pre-processing layer is shown in *yellow*, where the aim is to increase the noise power introduced by the steganography process and decrease the image content. In *green*, the convolutional layers appear where the hierarchical feature extraction is done. In *blue*, the functions of activation, scaling, absolute value layers, and normalization are observed. *White* shows the pooling operation that reduces the dimensionality of the feature map and the computational complexity. All the CNNs designed so far use Average Pooling operation due to the low power of the steganographic noise, which makes it necessary to take into account all the pixels of the region where the pooling operation will take place in order not to lose information. *Red* and *aquamarine green* show the classification module consisting of layers of neurons completely connected and a Softmax which is responsible of delivering a distribution of probabilities between 0 and 1 for each class defining whether the image is cover or stego.

The following information must be taken into account to read **Figure 6** correctly. The structure inside the boxes means

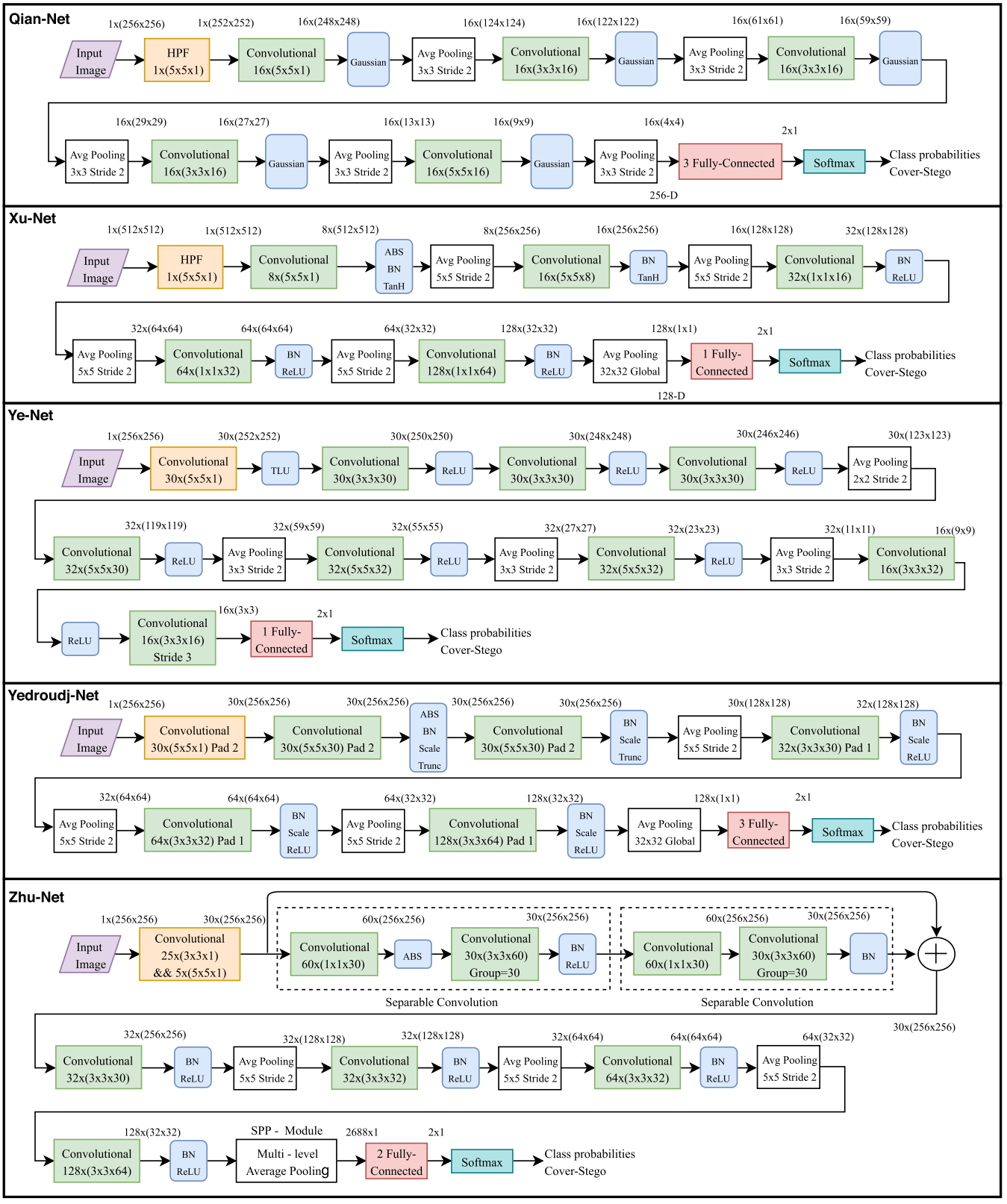

FIGURE 6. Most used CNNs architectures [25], [27], [38], [40], [51]. The data inside the boxes have the following structure: Number of kernels x (height x width x number of feature maps as input). The data outside the box has the following structure: Number of feature maps × (height × width). If the Stride or Padding is not specified, Stride =1 and Padding =0 are assumed.

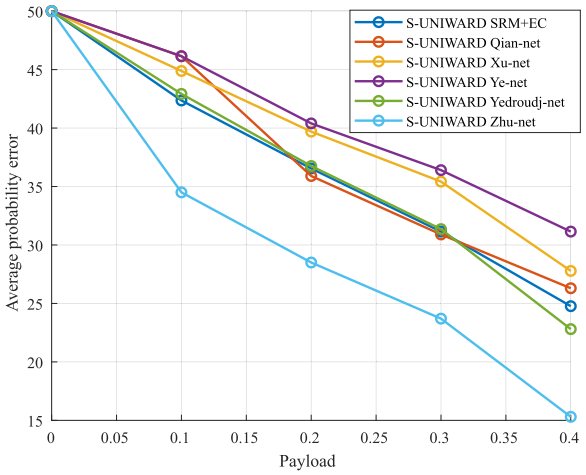

FIGURE 7. Average error percentage of different CNNs and SRMs using different payload values for the S-UNIWARD steganographic algorithm. [25], [27], [38], [40], [51].

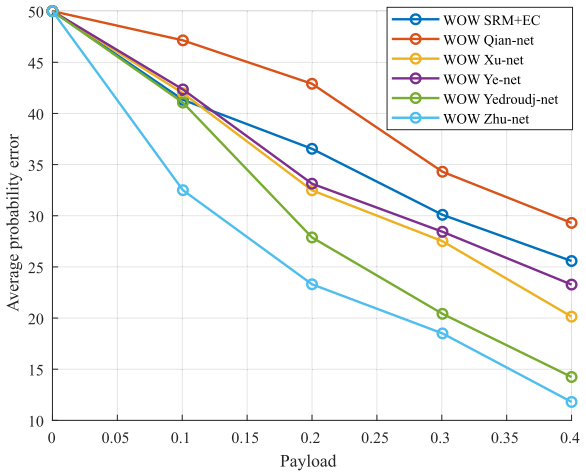

FIGURE 8. Average error percentage of different CNNs and SRMs using different payload values for the WOW steganographic algorithm. [25], [27], [38], [40], [51].

number of kernels × (height × width of the kernel × number of feature maps as an entry of the kernels). The structure to the outside of the boxes means number of feature maps × (height x width of the feature map). If Stride or Padding are not specified, Stride = 1 and Padding = 0 are assumed.

Figures 7 and **8** show the average error rate of steganographic images detection using the algorithms S-UNIWARD (fig 7) and WOW (fig 8) depending on the payload, for a range of 0bpp to 0.4bpp. It is important to note that as the payload increases the steganographic noise introduced in the image also increases, allowing CNNs to have more information to learn from this type of noise and consequently to improve the detection percentages. For the S-UNIWARD algorithm (fig 7) the lowest error percentage was obtained by **ZhuNet** regardless of the payload value, and observing specifically 0.4bpp (Most used payload by researchers), **ZhuNet** manages to decrease the error percentage by 7.5% compared to **YedroudjNet** (predecessor network) and by 9.4% compared to SRM+EC (traditional method). **YeNet** obtained the highest error rate during all payloads. It is important to note that **SRM+EC, QianNet, XuNet, YeNet** and **YedroudjNet** have similar behavior, which leaves **ZhuNet** as the first CNN to significantly exceed the detection percentages obtained by **SRM+EC** for the S-UNIWARD algorithm. For the WOW algorithm (fig 8) the lowest error percentage was obtained by **ZhuNet** regardless of the payload value, and specifically observing 0.4bpp, **ZhuNet** was able to decrease the error percentage by 2.3% compared to **YedroudjNet** and by 13.7% compared to **SRM+EC**. **QianNet** obtained the highest error rate during all payloads. It is important to point out that for the WOW algorithm the only CNN that did not exceed the **SRM+EC** results was **QianNet** (First CNN proposed), the other CNNs have exceeded the **SRM+EC** detection percentages.

The most used stenographic algorithms in the studied articles are S-UNIWARD, HUGO, HILL, WOW in the spatial domain and J-UNIWARD, UED, UERD in the frequency domain (JPEG), all with different payloads, usually the most used payload for experiments are 0.4 bpp for the spatial domain or 0.4 bpnzAC(bits per non-zero cover AC DCT coefficient) for the JPEG domain.

The comparison of the results of detection of steganographic images obtained by the proposed CNNs are made with respect to the traditional algorithms, which make a manual extraction of complex features, the most important algorithms are *SRM* [17], *SPAM* [24] and the variants for Selection-Channel-Aware [58], [59] for the spatial domain. Selection-Channel-Aware Gabor Filter Residuals (*SCA-GFR*) [47], [77], Discrete Cosine Transform Residual (*DCTR*) [78], JPEG Rich Models (*JRM*) [79], and PHase Aware pRojection Model (*PHARM*) [80] for frequency domain. The results obtained with the first CNNs were lower than those obtained by traditional algorithms, but as researchers advanced in the design of new networks or custom computational elements, the results of these CNNs outperformed the results reported in the literature. Most experiments use the Clairvoyant [26] scenario, which is characterized by as follows:

- The steganalyst knows the algorithm that was used to make the incrustation of the messages.
- The steganalyst has a good statistical knowledge distribution of the image databases used by the steganographer.
- The payload of the messages for the incrustation process is known.
- It is always worked with the same image size.
- The steganalyst has access to a set of cover-stego images with which the steganographer works.
- It works on the BOSSBase database of 10000 images with dimensions of 512 × 512 or 256 × 256 depending on the hardware available.
- From the BOSSBase initially of 10000 images (cover), other 10000 images are constructed with the incrustation

of messages by some of the existing steganographic algorithms (stego), in such a way that the complete set has 10000 pairs of images (cover-stego). From this set, 5000 pairs of images (cover-stego) are randomly selected, CNN is trained with 4000 pairs and validation is done with 1000 pairs, the remaining 5000 pairs of images are used for CNN evaluation.
- The initialization of the filter weights is done by the Xavier method [81].

The experiments mostly use the database BOSSBase V1.01 [12], [43] which consists of 10000 images in Portable Gray Map format (PGM) of 8 bits and size 512×512. The second database is the BOWS2 [82] which consists of 10000 images in PGM format of 8 bits and size 512×512. Finally, the third database is the extensive ImageNet [32], which is composed of more than 14 million images of different sizes. ImageNet database was normally employed for experiments conducted in the frequency domain (JPEG). In some experiments, the previous databases were resized or trimmed to 256×256 due to the computational cost and memory limitations of the researchers' teams.

Most used Frameworks for CNNs implementation are Cuda-convnet [83], Caffe [84] and TensorFlow [55], these toolboxes allow to create CNNs in a flexible and fast way. At Binghamton University (USA) [85] there is a large number of tools, such as algorithms for steganography and steganalysis (both in the spatial and frequency domain), traditional steganalyzers and applying DL techniques, digital image databases for experiments and some publications. Likewise, in the Laboratory of Informatics, Robotics and Microelectronics of the city of Montpellier-France (LIRMM) [86] there are several projects of DL applied to steganalysis, from which the algorithms can be downloaded, as well as the parameters of the CNNs trained by them and some important publications.

IV. CONCLUSIONS AND FUTURE WORK

A systematic review of DL applied to steganalysis, done in this paper, shows the evolution of the subject in a chronological way. Since 2014 the first CNN has been proposed with a stack of Auto-Encoders for unsupervised learning performed by *Tan and Li*, their results do not exceed those obtained by SRM, but they do surpass those obtained by SPAM, becoming a reasonable basis for other researchers. Since then, a great variety of research has been proposed, such as training sets of CNNs, transferring parameters from one network to another, quantitative steganalysis, steganalysis for arbitrary sized images, enrichment of image databases, taking into account the Cover-Source Mismatch effect in experiments, among others, which can be observed in detail in **Section III**.

Different architectures of CNNs are proposed to do steganalysis like the mentioned **QianNet**, **YuNet**, **YeNet**, **YedroudjNet**, **ZhuNet** all in the spatial domain, besides an adaptation of the CNN **YuNet** applying ResNet to do steganalysis in the frequency domain (JPEG). It is observed that the best detection results in the spatial domain are offered by CNN **ZhuNet** also improve the results offered by *SRM*. **SRNet** is a network proposal that avoids as much as possible the use of tricks or heuristics for the extraction process of the steganographic noise and works in the spatial domain and JPEG.

It is interesting to observe the implementation of the GAN methodology to make the automatic steganography process by placing two CNNs to compete with each other, one for steganography and the other for steganalysis.

Regarding the questions set forth in **Section II-B** we have shown that (1) detection performance obtained by DL applied to steganalysis has surpassed the results obtained by traditional methods (SRM+EC), as can be seen in **Figures 7**, **8** and **Tables 4**, **6**; (2) a variety of architectures address the specific challenge of steganalysis as shown in **Figure 6** and **Table 6**, furthermore, specific network components have been developed to address this challenge (such as TLU and Gaussian activation functions); (3) current detection levels are those reported in **Tables 4**, **6**, however they are yet far from results targeted by the research community, as conveyed in most of the literature, and specially in [34], [40], [42], [45], [56], [60], [87]; and (4) there is a limited but active set of databases for benchmarking steganalysis methods (see **Table 6**), however in the new challenge of steganalysis *ALASKA* [87] new real world oriented databases were released. These databases will serve as a basis for new experiments in steganography and steganalysis.

According to the present bibliographic review, we envisage possible future work as follows:
- Generate new CNNs unifying the advantages of existing networks or generate an entirely new architecture, (dense, shallow and/or deeper architectures), in order to improve the detection percentages, both in the spatial and frequency domain.
- Use different digital image databases, taking into account, for example, the use of different cameras, to test more experiments and study more deeply the Cover-Source Mismatch effect.
- Perform steganalysis by testing more steganographic algorithms in the JPEG domain.
- Adapt the GAN methodology to do steganalysis in the spatial domain and also use it to do automatic steganography in the JPEG domain.
- Adjust the CNNs that do quantitative steganalysis to improve your payload prediction results.
- Apply DL to quantitative steganalysis to predict the image steganographic payload in frequency domain (JPEG).
- Train existing CNNs with large scale databases and larger image sizes. In order to do this, it is necessary to do the training under a CPU and GPU cluster architecture in order to meet the demands of processing and memory.
- Train CNNs with a given steganographic algorithm and test on another algorithm to study how much transfer there can be from one algorithm to another.

- Apply the proposed ASDL-GAN framework to the JPEG domain, where millions of images are available, and incorporate more advanced deep learning architectures to improve its security performance.
- Generate new CNNs and design new computational elements that allow to obtain in a more efficient way the noise generated by the steganography process, to improve the representation of characteristics, to classify images in the spatial or frequency domain and to process arbitrary images, all of the above avoiding the use of tricks and in the most automatic way possible.
- Make a study of computational efficiency of the existing CNNs compared to traditional methods.
- Measure filters performance used in the pre-processing stage (HPF) in comparison with the activation functions used in DL applied to steganalysis.

  As can be seen above, there is a great variety of possibilities for future work that motivates researchers to continue contributing to this topic and invites new researchers to be interested in DL applied to steganalysis.

ACKNOWLEDGMENT

The authors would like to thank Simon Orozco-Arias, Laura Andrea Villada, Francy Nelly Jimenéz García, Omar Ángel Sánchez and Romain Guyot professors at the Universidad Autónoma de Manizales. They accompanied us in the process of writing and review, also we shared with these professors long talks about this topic. Likewise thanks to the project UN-UCALDAS Computational prototype for the fusion and analysis of large volumes of data in IoT (Internet of Things) environments, based on Machine Learning techniques and secure architectures Code: 36715, QUIPU code: 202010015371

REFERENCES

[1] N. Naranjo and V. Adolfo. (2007). *Estenografía en Contenido Multimedia*. [Online]. Available: http://openaccess.uoc.edu/webapps/o2/bitstream/10609/929/1/40114tfc.pdf
[2] *Crypto Law Survey*. Accessed: Dec. 15, 2018. [Online]. Available: http://www.cryptolaw.org
[3] G. J. Simmons, "The prisoners' problem and the subliminal channel," in *Advances in Cryptology*, D. Chaum, Ed. Boston, MA, USA: Springer, 1984, pp. 51–67. doi: 10.1007/978-1-4684-4730-9_5.
[4] (2015). *Aplicaciones de la esteganografía en la seguridad informática*. [Online]. Available: https://www.sans.org/reading-room/whitepapers/stenganography/hiding-plain-view-steganography-terrorist-tool-551
[5] N. F. Johnson and S. Jajodia, "Exploring steganography: Seeing the unseen," *Computer*, vol. 31, no. 2, pp. 26–34, Feb. 1998. doi: 10.1109/MC.1998.10029.
[6] J. Fridrich, M. Goljan, and R. Du, "Detecting LSB steganography in color, and gray-scale images," *IEEE Multimedia Mag.*, vol. 8, no. 4, pp. 22–28, Oct./Dec. 2001.
[7] T. Pevný, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Information Hiding*, R. Böhme, P. W. L. Fong, and R. Safavi-Naini, Eds. Berlin, Germany: Springer, 2010, pp. 161–177.
[8] B. Li, M. Wang, J. Huang, and X. Li, "A new cost function for spatial image steganography," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 4206–4210.
[9] V. Sedighi, R. Cogranne, and J. Fridrich, "Content-adaptive steganography by minimizing statistical detectability," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 2, pp. 221–234, Feb. 2016.
[10] V. Holub, J. Fridrich, and T. Denemark, "Universal distortion function for steganography in an arbitrary domain," *EURASIP J. Inf. Secur.*, vol. 2014, p. 1, Dec. 2014. doi: 10.1186/1687-417X-2014-1.
[11] V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2012, pp. 234–239.
[12] *BOSS*. Accessed: Dec. 15, 2018. [Online]. Available: http://agents.fel.cvut.cz/boss/index.php?mode=VIEW&tmpl=materials
[13] Y. J. Chanu, K. M. Singh, and T. Tuithung, "Image steganography and steganalysis: A survey," *Int. J. Comput. Appl.*, vol. 52, no. 2, pp. 1–11, 2012.
[14] A. Westfeld, "F5—A steganographic algorithm," in *Information Hiding*, I. S. Moskowitz, Ed. Berlin, Germany: Springer, 2001, pp. 289–302.
[15] L. Guo, J. Ni, and Y. Q. Shi, "Uniform embedding for efficient JPEG steganography," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 5, pp. 814–825, May 2014.
[16] L. Guo, J. Ni, W. Su, C. Tang, and Y.-Q. Shi, "Using statistical image model for JPEG steganography: Uniform embedding revisited," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 12, pp. 2669–2680, Dec. 2015.
[17] J. Fridrich and J. Kodovský, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 868–882, Jun. 2012.
[18] J. Kodovský, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 432–444, Apr. 2012.
[19] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, 2011. [Online]. Available: http://doi.acm.org/10.1145/1961189.1961199
[20] I. Lubenko and A. D. Ker, "Steganalysis with mismatched covers: Do simple classifiers help?" in *Proc. Multimedia Secur.*, New York, NY, USA, 2012, pp. 11–18. [Online]. Available: http://doi.acm.org/10.1145/2361407.2361410
[21] M. A. Nielsen, "Neural networks and deep learning," *Mach. Learn.*, pp. 875–936, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128015223000185
[22] R. T. Soto. (2016). *Programación Paralela Sobre Arquitecturas Heterogéneas*. [Online]. Available: http://www.bdigital.unal.edu.co/54267/
[23] S. Tan and B. Li, "Stacked convolutional auto-encoders for steganalysis of digital images," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA)*, no. 1, Dec. 2014, pp. 1–4. [Online]. Available: https://ieeexplore.ieee.org/document/7041565
[24] T. Pevný, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 215–224, Jun. 2010.
[25] Y. Qian, J. Dong, W. Wang, and T. Tan, "Deep learning for steganalysis via convolutional neural networks," *Proc. SPIE*, vol. 9409, Mar. 2015, Art. no. 94090J. [Online]. Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/9409/94090J/Deep-learning-for-steganalysis-via-convolutional-neural-networks/10.1117/12.2083479.short. doi: 10.1117/12.2083479.
[26] L. Pibre, P. Jérôme, D. Ienco, and M. Chaumont, "Deep learning is a good steganalysis tool when embedding key is reused for different images, even if there is a cover source-mismatch," in *Proc. Media Watermarking, Secur., Forensics*, San Francisco, CA, USA, Feb. 2016, pp. 14–18. [Online]. Available: http://arxiv.org/abs/1511.04855
[27] G. Xu, H.-Z. Wu, and Y.-Q. Shi, "Structural design of convolutional neural networks for steganalysis," *IEEE Signal Process. Lett.*, vol. 23, no. 5, pp. 708–712, May 2016.
[28] G. Xu, H.-Z. Wu, and Y. Q. Shi, "Ensemble of CNNs for steganalysis: An empirical study," in *Proc. 4th ACM Workshop Inf. Hiding Multimedia Secur.*, 2016, pp. 103–107. [Online]. Available: http://doi.acm.org/10.1145/2909827.2930798
[29] Y. Qian, J. Dong, W. Wang, and T. Tan, "Learning and transferring representations for image steganalysis using convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 2752–2756.
[30] J. Zeng, S. Tan, B. Li, and J. Huang, "Large-scale JPEG image steganalysis using hybrid deep-learning framework," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 5, pp. 1200–1214, May 2018.
[31] J. Zeng, S. Tan, B. Li, and J. Huang, "Pre-training via fitting deep neural network to rich-model features extraction procedure and its effect on deep learning for steganalysis," *Electron. Imag.*, vol. 6, no. 7, pp. 44–49, 2017.
[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, vol. 1. New York, NY, USA: Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://dl.acm.org/citation.cfm?id=2999134.2999257

[33] G. Xu, "Deep convolutional neural network to detect J-UNIWARD," in *Proc. 5th ACM Workshop Inf. Hiding Multimedia Secur.*, 2017, pp. 67–73. [Online]. Available: http://doi.acm.org/10.1145/3082031.3083236

[34] M. Boroumand, M. Chen, and J. Fridrich, "Deep residual network for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 5, pp. 1181–1193, May 2018.

[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn.*, vol. 37, 2015, pp. 448–456. [Online]. Available: http://arxiv.org/abs/1502.03167 and http://dl.acm.org/citation.cfm?id=3045118.3045167

[36] D. Hu, L. Wang, W. Jiang, S. Zheng, and B. Li, "A novel image steganography method via deep convolutional generative adversarial networks," *IEEE Access*, vol. 6, pp. 38303–38314, 2018.

[37] W. Tang, S. Tan, B. Li, and J. Huang, "Automatic steganographic distortion learning using a generative adversarial network," *IEEE Signal Process. Lett.*, vol. 24, no. 10, pp. 1547–1551, Oct. 2017.

[38] J. Ni, J. Ye, and Y. I. Yang, "Deep learning hierarchical representations for image steganalysis," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 11, pp. 2545–2557, Nov. 2017.

[39] K. He and X. Zhang and S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[40] M. Yedroudj, F. Comby, and M. Chaumont, "Yedroudj-net: An efficient CNN for spatial steganalysis," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2018, pp. 2092–2096. [Online]. Available: http://arxiv.org/abs/1803.00407

[41] C. F. Tsang and J. J. Fridrich, "Steganalyzing images of arbitrary size with CNNs," in *Proc. Media Watermarking, Secur., Forensics*, Burlingame, CA, USA, Jan. /Feb. 2018, pp. 121-1–121-8. doi: 10.2352/ISSN.2470-1173.2018.07.MWSF-121.

[42] M. Yedroudj, M. Chaumont, and F. Comby, "How to augment a small learning set for improving the performances of a CNN-based steganalyzer?" *Electron. Imag.*, vol. 7, pp. 1–7, Jan. 2018. [Online]. Available: http://arxiv.org/abs/1801.04076

[43] P. Bas, T. Filler, and T. Pevny, "'Break our steganographic system': The ins and outs of organizing BOSS," in *Information Hiding* (Lecture Notes in Computer Science), vol. 6958. Czechia: May 2011, pp. 59–70. [Online]. Available: https://hal.archives-ouvertes.fr/hal-00648057, doi: 10.1007/978-3-642-24178-9_15.

[44] W. Mazurczyk and S. Wendzel, "Information hiding: Challenges for forensic experts," *Commun. ACM*, vol. 61, no. 1, pp. 86–94, 2017. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3176926.3158416. doi: 10.1145/3158416.

[45] M. Chen, M. Boroumand, and J. Fridrich, "Deep learning regressors for quantitative steganalysis," *Soc. Imag. Sci. Technol.*, vol. 2018, no. 7, pp. 160-1–160-7, 2017.

[46] B. Li, W. Wei, A. Ferreira, and S. Tan, "ReST-Net: Diverse activation modules and parallel subnets-based CNN for spatial image steganalysis," *IEEE Signal Process. Lett.*, vol. 25, no. 5, pp. 650–654, May 2018.

[47] X. Song, F. Liu, C. Yang, X. Luo, and Y. Zhang, "Steganalysis of adaptive JPEG steganography using 2D Gabor filters," in *Proc. 3rd ACM Workshop Inf. Hiding Multimedia Secur.*, New York, NY, USA, 2015, pp. 15–23. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2756601.2756608 and http://doi.acm.org/10.1145/2756601.2756608

[48] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Int. Conf. Mach. Learn. (ICML)*, no. 3. Madison, WI, USA: Omnipress, 2010, pp. 807–814. [Online]. Available: http://dl.acm.org/citation.cfm?id=3104322.3104425

[49] B. Karlik and A. Vehbi, "Performance analysis of various activation functions in generalized MLP architectures of neural networks," *Int. J. Artif. Intell. Expert Syst.*, vol. 1, no. 4, pp. 111–122, 2015.

[50] J. Zeng, S. Tan, G. Liu, B. Li, and J. Huang, "WISERNet: Wider separate-then-reunion network for steganalysis of color images," Mar. 2018, *arXiv:1803.04805*. [Online]. Available: http://arxiv.org/abs/1803.04805

[51] R. Zhang, F. Zhu, J. Liu, and G. Liu, "Efficient feature learning and multi-size image steganalysis based on CNN," Jul. 2018, *arXiv:1807.11428*. [Online]. Available: http://arxiv.org/abs/1807.11428

[52] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Computer Vision—ECCV*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham, Switzerland: Springer, 2014, pp. 346–361.

[53] K. Karampidis, E. Kavallieratou, and G. Papadourakis, "A review of image steganalysis techniques for digital forensics," *J. Inf. Secur. Appl.*, vol. 40, pp. 217–235, Jun. 2018. doi: 10.1016/j.jisa.2018.04.005.

[54] Software Engineering Group, "Guidelines for performing systematic literature reviews in software engineering," Softw. Eng. Group School Comput. Sci., Math. Keele Univ., Keele, U.K., Dept. Comput. Sci. Univ. Durham, Durham, U.K., EBSE Tech. Rep. EBSE-2007-01, 2007.

[55] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," in *Proc. 12th USENIX Conf. Oper. Syst. Design Implement.*, vol. 101. Berkeley, CA, USA: USENIX Association, 2016, pp. 265–283. [Online]. Available: http://dl.acm.org/citation.cfm?id=3026877.3026899

[56] M. Chaumont, "The emergence of Deep Learning in steganography and steganalysis," in *Journée 'Stéganalyse, Enjeux et Méthodes', labelisée par le GDR ISIS et le pré-GDR sécurité*. Poitiers, France: Philippe Carré (XLIM, Poitiers) and Marianne Clausel (IECL,Nancy) and Farida Enikeeva (LMA, Poitiers) and Laurent Navarro (CIS-EMSE, St Etienne), Jan. 2018. [Online]. Available: https://hal-lirmm.ccsd.cnrs.fr/lirmm-01777391. doi: 10.13140/RG.2.2.35080.32005.

[57] S. Kouider, M. Chaumont, and W. Puech, "Technical points about adaptive steganography by Oracle (ASO)," in *Proc. 20th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2012, pp. 1703–1707.

[58] L. Pibre, P. Jérôme, D. Ienco, and M. Chaumont, "Deep learning is a good steganalysis tool when embedding key is reused for different images, even if there is a cover source-mismatch," in *Media Watermarking, Security, and Forensics*. San Francisco, CA, USA: IS&T Int. Symp. on Electronic Imaging, 2016, pp. 1–23. [Online]. Available: http://www.lirmm.fr/~chaumont/

[59] J. Newman, G. Yong, W. Min, R. Stephanie, and L. Li, "StegoDB: A statistically-designed image dataset for benchmarking steganalysis algorithms WeHide app," in *Forensics@NIST*. Ames, IA, USA: Iowa State Univ., 2016.

[60] M. Chaumont. (Oct. 2018). *Deep Learning in Steganography and Steganalysis Since 2015*. [Online]. Available: http://rgdoi.net/10.13140/RG.2.2.25683.22567

[61] M. Chen, V. Sedighi, M. Boroumand, and J. Fridrich, "JPEG-phase-aware convolutional neural network for steganalysis of JPEG images," in *Proc. 5th ACM Workshop Inf. Hiding Multimedia Secur. (IHMMSec)*, 2017, pp. 75–84. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3082031.3083248 and http://doi.acm.org/10.1145/3082031.3083248

[62] J. Yang, X. Kang, E. K. Wong, and Y.-Q. Shi, "JPEG steganalysis with combined dense connected CNNs and SCA-GFR," *Multimedia Tools Appl.*, vol. 78, no. 7, pp. 8481–8495, 2019.

[63] X. Huang, S. Wang, T. Sun, G. Liu, and X. Lin, "Steganalysis of adaptive JPEG steganography based on resdet," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, Nov. 2018, pp. 549–553.

[64] J. Yang, Y.-Q. Shi, E. K. Wong, and X. Kang, "JPEG steganalysis based on DenseNet," 2017, *arXiv:1711.09335*. [Online]. Available: https://arxiv.org/abs/1711.09335

[65] S. Wu, S. Zhong, and Y. Liu, "Deep residual learning for image steganalysis," *Multimedia Tools Appl.*, vol. 77, no. 9, pp. 10437–10453, 2017.

[66] S. Wu, S.-H. Zhong, and Y. Liu, "Steganalysis via deep residual network," in *Proc. 22nd Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Dec. 2016, pp. 1233–1236.

[67] Y. Zhang, W. Zhang, K. Chen, J. Liu, Y. Liu, and N. Yu, "Adversarial examples against deep neural network based steganalysis," in *Proc. 6th ACM Workshop Inf. Hiding Multimedia Secur.*, New York, NY, USA, 2018, pp. 67–72. [Online]. Available: http://doi.acm.org/10.1145/3206004.3206012

[68] J. Hayes and G. Danezis, "Generating steganographic images via adversarial training," 2017, *arXiv:1703.00371*. [Online]. Available: https://arxiv.org/abs/1703.00371

[69] J. Yang, K. Liu, X. Kang, E. K. Wong, and Y.-Q. Shi, "Spatial image steganography based on generative adversarial network," 2018, *arXiv:1804.07939*. [Online]. Available: https://arxiv.org/abs/1804.07939

[70] W. Tang, B. Li, S. Tan, M. Barni, and J. Huang, "CNN based adversarial embedding with minimum alteration for image steganography," 2018, *arXiv:1803.09043*. [Online]. Available: https://arxiv.org/abs/1803.09043

[71] J. Liu *et al.*, "Detection based defense against adversarial examples from the steganalysis point of view," 2018, *arXiv:1806.09186*. [Online]. Available: https://arxiv.org/abs/1806.09186

[72] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, "HiDDeN: Hiding data with deep networks," 2018, *arXiv:1807.09937*. [Online]. Available: https://arxiv.org/abs/1807.09937

[73] A. Zakaria, M. Chaumont, and G. Subsol, "Quantitative and binary steganalysis in JPEG: A comparative study," in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, vol. 201, no. 1, 2018, pp. 1422–1426.

[74] M. Yedroudj, M. Chaumont, and F. Comby, "How to augment a small learning set for improving the performances of a CNN-based steganalyzer?" Jan. 2018, *arXiv:1801.04076*. [Online]. Available: https://arxiv.org/abs/1801.04076
[75] Y.-L. Boureau, J. Ponce, and Y. LeCun, "A theoretical analysis of feature pooling in visual recognition," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*. New York, NY, USA: Omnipress, 2010, pp. 111–118. [Online]. Available: http://dl.acm.org/citation.cfm?id=3104322.3104338
[76] Y. Qian, J. Dong, W. Wang, and T. Tan, "Feature learning for steganalysis using convolutional neural networks," *Multimedia Tools Appl.*, vol. 77, no. 15, pp. 19633–19657, Aug. 2018. [Online]. Available: http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/1%2.2083479
[77] T. D. Denemark, M. Boroumand, and J. Fridrich, "Steganalysis features for content-adaptive JPEG steganography," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 8, pp. 1736–1746, Aug. 2016.
[78] V. Holub and J. Fridrich, "Low-complexity features for JPEG steganalysis using undecimated DCT," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 2, pp. 219–228, Feb. 2015.
[79] J. Kodovský and J. Fridrich, "Steganalysis of JPEG images using rich models," *Proc. SPIE*, vol. 8303, Feb. 2012, Art. no. 83030A. doi: 10.1117/12.907495.
[80] V. Holub and J. Fridrich, "Phase-aware projection model for steganalysis of JPEG images," p. 94090T, 2015. [Online]. Available: http://10.0.4.93/12.2075239
[81] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *J. Mach. Learn. Res.*, vol. 9, pp. 249–256, May 2010. [Online]. Available: http://arxiv.org/abs/1701.05369
[82] (2007). *BOWS-2 Web Page*. [Online]. Available: http://bows2.ec-lille.fr/index.php?mode=VIEW&tmpl=index1
[83] *Google Code Archive—Long-Term Storage for Google Code Project Hosting*. Accessed: Dec. 15, 2018. [Online]. Available: https://code.google.com/archive/p/cuda-convnet/
[84] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22Nd ACM Int. Conf. Multimedia (MM)*, New York, NY, USA, 2014, pp. 675–678. [Online]. Available: http://doi.acm.org/10.1145/2647868.2654889
[85] *Universidad de Binghamton*. [Online]. Available: http://dde.binghamton.edu/download/
[86] *Page d'Accueil de Marc Chaumont*. Accessed: Dec. 15, 2018. [Online]. Available: http://www.lirmm.fr/~chaumont/index.html
[87] R. Cogranne, Q. Giboulot, and P. Bas, Alaska. *Marc Chaumont's Web Page*. Accessed: Jan. 19, 2019. [Online]. Available: https://alaska.utt.fr/#top
[88] D. M. Chandler and S. S. Hemami, "VSNR: A wavelet-based visual signal-to-noise ratio for natural images," *IEEE Trans. Image Process.*, vol. 16, no. 9, pp. 2284–2298, Sep. 2007.
[89] R. Shah and Y. Yang, "Health and economic burden of obesity in elderly individuals with asthma in the United States," *Population Health Manage.*, vol. 18, no. 3, pp. 186–191, 2015. [Online]. Available: http://online.liebertpub.com/doi/10.1089/pop.2014.0089
[90] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. 30th IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1800–1807.

TABARES-SOTO REINEL received the B.S. degree in electronic engineering from the National University of Colombia, in 2009, the B.S. degree from the Universidad de Caldas, Colombia in systems and computer engineering, in 2016, and the M.S. degree in engineering from the National University of Colombia, in 2017. He is currently pursuing the Ph.D. degree in computer science with the Universidad Autónoma de Manizales, Colombia, where he has been a Professor and a Coordinator with the Department of Electronics, since 2014. His main research interests include steganalysis, machine learning, deep learning, bioinformatics, and high-performance computing.

RAMOS-POLLÁN RAÚL received the Ph.D. degree in computer engineering (analysis of biomedical images) from the University of Porto, Portugal. He is a Professor and a Researcher in artificial intelligence and machine learning with the Universidad de Antioquia. He has developed his career in both industry and academia, working as the Director of the CETA Computing Center, CIEMAT, Extremadura, Spain, a Software Engineer with the European Center for Particle Physics, a Java Architect with Sun Microsystems Switzerland, and the Co-Founder of Pildo Labs, an SME in aeronautics and software sector based in Barcelona. He moved to Colombia, in 2012, initially as a Guest Researcher in image analytics with the MindLab Group, National University of Colombia, Bogotá, moved to the University Industrial of Santander, in 2013, and most recently moved to his current position with the Universidad de Antioquia, in 2018.

ISAZA GUSTAVO received the B.S. degree in system and computing engineering from the Autonomous University of Manizales, Colombia, in 1997, the master's degree in networking software development from Andes University, Colombia, in 1998, and the M.Sc./DEA and Ph.D. degrees from the Pontifical University of Salamanca, Spain, in 2008 and 2010, respectively. He is a Full Professor/Senior Researcher with the Universidad de Caldas and a member of the GITIR Research Group. He has published over 40 papers and conferences related to machine learning in cybersecurity, bioinformatics, AI in videogames, and distributed computing.