Efficient Blind Detection of JPEG Steganography

Johann Barbier[1,2], Éric Filiol[2], Kichenakoumar Mayoura[2]

[1] CELAR, Département de Cryptologie, La Roche Marguerite, BP 57419, 35174 Bruz Cedex, France
[2] ESAT, Laboratoire de Virologie et Cryptologie, BP 18, 35998 Rennes Cedex, France
Email: johann.barbier@dga.defense.gouv.fr, {eric.filiol, anada.mayoura}@esat.terre.defense.gouv.fr

Abstract— In this paper, we present a novel approach which consists in exploring statistics in the compressed frequency domain. This approach is motivated by two main characteristics of the lossless compression step of the JPEG format. First, this step can be viewed as a bijective mapping and then, it has an avalanche criterion close to 0.5. This properties enable the design of detection schemes which the efficiencies are constant and do not depend on the amount of information that has been embedded. This characteristics define a new class of promising functions for steganalysis. We illustrate our technique by considering RLE plus Huffman as such a function and evaluate it by blindly detecting the use of Outguess, F5 and JPHide and JPseek. Experimental results show that our steganalysis scheme is able to efficiently detect the use of new algorithms which are not used in the training step, even if the embedding rate is very low ($\approx 10^{-6}$). As expected, the accuracy of our detector is independent of the payload.

Index Terms— universal steganalysis, JPEG, Kullbak-Leibler distance, Fisher discriminant.

I. INTRODUCTION

Steganography is the science of *covered writing*. Its purpose is to hide information in a cover medium so that it is "hard" for everyone to detect the existence of the embedded information. On the opposite side, steganalytic schemes tend to detect hidden information in a mass of cover media. Let Alice and Bob communicate using a steganographic algorithm $\mathcal{A}$, for instance, to hide the world their love affair, and Eve, the paparazi, who will earn lot of money if she can prove Alice and Bob are lovers. In a classical model and according to the Kerchoff's principles, Eve knows all the steganographic techniques Alice and Bob are likely to use. So, she can design dedicated methods to detect the use of $\mathcal{A}$ specifically; this is called *specific steganalysis*. In a harder model of attack, we make the hypothesis that Alice and Bob keep their steganographic algorithm secret and Eve does not know the specifications of $\mathcal{A}$. Her goal is now to build a detector, which does not depend on $\mathcal{A}$ and which distinguishes cover and stego media in order to prove that Alice and Bob indeed share secrets through steganography; this is called *universal steganalysis*.

Specific and universal steganalysis do not achieve the same goal; the specific steganalysis answers the question: "*Is the medium was embedded with the algorithm $\mathcal{A}$?*" and the

universal steganalysis answers the question : "*Is the medium a stego medium ?*". Even if the universal steganalysis is more general and so less efficient than the specific one for detecting the use of a given steganographic algorithm, there are two main interests for using it. First, universal steganalysis schemes are independent of the steganographic algorithms; stego media embedded with an *unknown* algorithm may also be detected by such schemes. Secondly, it is the only possible way to detect the use of steganographic algorithm for which no specific steganalysis is known. So, the central property universal steganalysis schemes should verify is the following one: given a set of known steganographic algorithms for the training step, we are able to detect the use a new steganographic algorithm which is not in the previous set. If it is not the case, the considered scheme is vulgarly dependent on a steganographic algorithm and is a specific steganalysis scheme. In the remaining of this paper this main property will be called *universality property*. We also propose to extend the definition of universal steganalysis to a stronger concept of *unconditional steganalysis*. A steganalytic scheme would be said *unconditional* if and only if, given a set of known steganographic algorithms, the scheme is able to detect the use of *any* new steganographic algorithm which is not in the previous set. The universal steganalysis will be studied through the scope of the universality property and the efficiency of the proposed scheme will be measured by the detection rates when detecting algorithms which are not in the training set.

In this paper, we take place in Eve's shoes, and our goal is to detect the existence of embedded message into JPEG images. The training set of our universal steganalytic scheme is composed of images embbeded with the well known steganographic algorithms, Outguess [1], F5 [2] and JPHide [3] but it can also be designed with another algorithms in the same way. In JPEG steganalysis, people traditionally try to find detectable properties directly studying statistics of the DCT coefficients or of the decompressed images. By contrast, we propose to examine Huffman compressed data, which are DCT coefficients compressed first by RLE and then by Huffman compression algorithms. We point out new statistical features to detect hidden information in JPEG images. We examine different cases of training sets and evaluate the universality property of our scheme.

The starting point of our work was presented in [4]. We detail here some keys to deeply understand why the experimental results pass beyound the limit set by the compromise between

the capacity and the detection of a steganography scheme. This paper suggests the existence of a particular group of mappings which could be explored to conceive steganalysis techniques which the accuracies do not depend on the payload. We also discuss the way to correctly code our algorithms and their computational limits.

In the first section, we quickly present the JPEG standard, the DCT-based steganography and give a brief description of the steganography algorithms we blindly detect. In the second section, we recall state of the art JPEG steganalysis techniques, put our approach back in its place and connect it to our previous works. We also present a new approach for JPEG steganalysis based on statistics in the compressed frequency domain and point out some function characteristics in order to define a new class of good functions for steganalysis. Then, we present our statistical features and how we chose them. In section **IV**, we explain the design of our Fisher classifier, detail the experimental framework and the results we obtained. We evaluate the efficiency of the scheme using Outguess, F5, and JPHide algorithms. Some keys are also given in order to code the proposed steganalysis scheme. Finally, we conclude in the last section and give some discussions.

II. JPEG STEGANOGRAPHY

A. The JPEG Format

The Joint Photographic Expert Group (JPEG) was created in 1986. This Group worked on digital compression and coding of continuous-tone still images. These studies have led to the CCITT[1] recommendation T.81 and the ISO[2] Standard 10918-1.

The JPEG format defines four types of compression modes which are sequential, progressive, hierarchical and lossless. In our case, the progressive mode is used.

DCT[3]-Based Coding: The figure 1 explains the main procedures for all encoding processes based on the DCT. In order to simplify, the diagram operates on a single-component image.
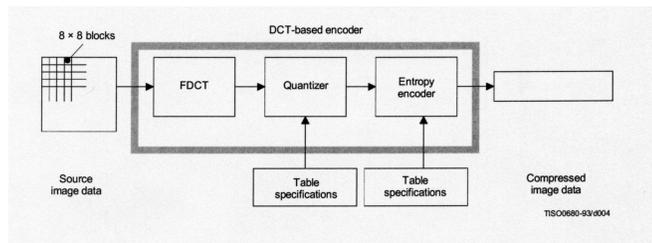

Figure 1. DCT-based encoder simplified diagram

Main Characteristics of Coding Processes: A digital image can be represented by pixels. The three color coefficients (Red, Green, Blue or RGB) for each pixel are transformed into a new coding scheme: one luminance coefficient (Y) and two chrominance coefficients (U and V or also called Cb and Cr).

After the conversion from RGB to YCbCr, the values, are gouped in 8×8 pixels blocks, and transformed by a forward DCT. Most of the frequency coefficients obtained are very low and we can remove a lot of them and still reconstruct the original values. The low frequencies are conserved while the high frequencies are removed.

After the DCT transformation on each block, the DCT coefficients are quantized. This step called quantization is the main lossy process. The coefficients are divided with fixed values coming from a specified table and then rounded. Most of the quantized DCT coefficients are equal to zero.

The "zig-zag" order consists to order the coefficients in each 8×8 block (most of them are equal to zero).

After the "zig-zag" sequence, the last steps are lossless compression. First a simple RLE[4] is used to compress the high frequency coefficients. Then a Huffman coding procedure is applied. Finally, the output is the JPEG raw binary data.

B. Embedding Information in the DCT Coefficients

The JPEG compression process can be divided into two main parts: the first one computes quantized DCT coefficients from a bitmap image $\mathcal{B}$ and some parameters $\mathcal{P}_1$; it will be noted $\mathcal{C}_l$.

$$\mathcal{C}_l : (\mathcal{B}, \mathcal{P}_1) \longrightarrow (DCT_i), \text{ where } DCT_i \in \mathbb{Z}.$$

$\mathcal{C}_l$ is a lossy compression, that means $\mathcal{C}_l$ is not a bijective mapping. So, if we apply $\mathcal{D}_l$, the decompression algorithm associated to $\mathcal{C}_l$ we don't retrieve $\mathcal{B}$.

$$\mathcal{D}_l : ((DCT_i), \mathcal{P}_1) \longrightarrow \mathcal{B}' \text{ with } \mathcal{B}' \neq \mathcal{B}.$$

The second one computes a string of binary compressed data from quantized DCT coefficients and some parameters $\mathcal{P}_2$; it will be noted $\mathcal{C}_u$.

$$\mathcal{C}_u : ((DCT_i), \mathcal{P}_2) \longrightarrow (b_j) \text{ where } b_i \in \mathbb{F}_2.$$

$\mathcal{C}_u$ is an unlossy compression, that implies it is a bijective mapping.

Since $\mathcal{C}_l$ is not a bijective mapping, one cannot naturally hide information during the first step, otherwise some of the embedded information will not be retrieved. Information can only be hidden during the second step. This step, as we saw previously, is divided into zig-zag re-ordering, RLE and Huffmann encodings. So, the only practical way to embed any information is in DCT coefficients, after RLE or Huffmann encodings. To minimize the distortions of the original image, DCT are the most adapted.

---

[1] International Telegraph and Telephone Consultative Committee
[2] International Standard Organisation
[3] Discrete Cosine Transform

[4] Run Length Encoding

The main problem, when embedding information in DCT coefficients, is to preserve the statistics of the cover medium. State of the art steganographic systems take care of keeping DCT statistics unchanged, histogram for example, but even if DCT statistics are preserved, many steganalysis schemes [5], [6], [7], [8], [9] are based on deviations of some decompressed cover image statistics. It seems that both cannot be preserved at the same time.

C. Detected Steganographic Algorithms

1) The Outguess: The Outguess steganographic algorithm [1] was proposed by N. Provos in 2001. It was designed to preserve first-order statistics. Outguess embeds information in two main steps as follows. First, using a RC4 based PRGN, the algorithm embeds message bits into randomly choosen redunctant LSB of the DCT coefficients. Then, in a second step, some LSB of DCT coefficients are flipped in order that the DCT histogram of the stego image is as close as possible to the DCT histogram of the cover image.

2) F5: The F5 steganographic algorithm [2] was proposed by A. Westfeld in 2001. As Outguess, it is designed to preserve first order statistics, notably the DCT histogram. First, F5 permutes all DCT coefficients using a PRNG. Then, it encodes the message with an error correcting code and embeds the associated code words with introduced well choosen errors, into non zero DCT coefficients. By this way, F5 increases the capacity of the cover image. Unlike Outguess, F5 does not use the LSB of the DCT coefficients but decreases the absolute values of non zero DCT coefficients. The algorithm was designed to prevent its use from χ^2 steganalysis [10] by mapping the DCT values to the steganographic values: even negative and odd positive coefficients embed an one value, then odd negative and even positive ones embed a zero value.

3) JPHide and JPSeek: JPHide is a steganographic system developped by A. Latham in 1999 [3] which embeds data in LSB of the DCT coefficients. It uses a PRNG based on Blowfish.

III. DETECTING JPEG STEGO IMAGES

A. JPEG Steganalysis Methods

Different approaches have been used to detect stego images. The first one consists in studying directly DCT coefficients like J. Fridich [11], [12] who looked at first order statistics and at the discontinuity of DCT coefficients at the borders of blocks for detecting the use of F5 and Outguess. She also pointed out some other features for the frequency domain [13], [14] for JPEG syteganalysis.

The second approach is dedicated to the spatial domain. H. Farid and S. Lyu obtained classifier with a high detection rate by combining Support Vector Machines (SVM) with higher order statistics [6], [7] or with wavelet transform statistics [8], [9] of decompressed JPEG image. J. J. Harmsen et al. [15] proposed to use a Fisher discriminant instead of a SVM and

I. Avicib et al. [5] introduced metrics based on images quality.

Previous methods have even been used together [16] to increase the accuracy of detectors. Among these techniques we can distinguish two categories of steganalysis: *specific steganalysis* and *universal steganalysis*.

1) Specific Steganalysis: Specific steganalysis is dedicated to only a given embedding algorithm. It may be very accurate for detecting images embedded with the given steganographic algorithm but it fails to detect those embedded with another algorithm. Techniques developed in [11], [12], [13], [15], [17] are specific.

2) Universal Steganalysis: Universal steganalysis enables to detect stego images whatever the steganographic system be used. Because it can detect a larger class of stego images, it is generally less accurate for one given steganographic algorithm. Methods presented in [5], [6], [14], [16], [7], [8], [9], [4] are universal.

In this paper, we will study an universal method adapted for the compressed frequency domain. Our novel approach consists in focusing on this domain, whereas up to now all the steganalysis schemes deal with the spatial or the frequency domain. But, unlike classical universal steganalysis schemes, the main point we want to show is the ability of our universal steganalyzer to detect the use of a steganography algorithms not used during the training step. The first results of this work has been presented in [4]. In the same spirit and taking advantages of the easy way to detect statistical deviations in the compressed frequency domain, we designed the Multiple Embedding Method [17] for specific JPEG steganalysis and obtained better results in specifically detecting Outguess, F5 and JPHide and JPSeek. In the same way, our specific detectors have detection rates independent of the embedding rate and so are able to detect JPEG image embedded with only few bytes.

B. A New Point of View

We have to keep in mind three important intuitive assertions:
- embedding information in DCT_i, will change $\mathcal{D}_l((DCT_i), \mathcal{P}_1)$ but also $\mathcal{C}_u((DCT_i), \mathcal{P}_2)$.
- one cannot preserve at the same time the statistics of DCT_i, those of $\mathcal{D}_l((DCT_i), \mathcal{P}_1)$ and $\mathcal{C}_u((DCT_i), \mathcal{P}_2)$.
- hiding information tends to introduce a variation of entropy.

Most of steganalytic techniques consist in observing some statistical deviations directly on DCT coefficients or in $\mathcal{D}_l((DCT_i), \mathcal{P}_1)$. We propose here to explore statistics in $\mathcal{C}_u((DCT_i), \mathcal{P}_2)$.

Let I a given JPEG image to analyse and $(b_j)^5$ the output of $\mathcal{C}_u$. We noticed a variation of the entropy of the output stream

---

[5](b_j) is only composed of the RLE and Huffman compressed DCT coefficients and does not include the JPEG file header.
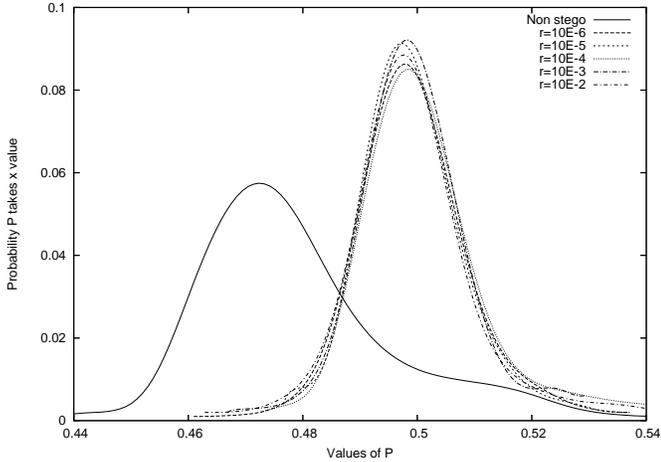
Figure 2. Density probability functions of P for JPHide no stego and stego images with different embedding rates r.

Figure 3. Avalanche criterion of RLE+Huffman compression function.

when the image has been embedded with a steganographic scheme. The binary entropy $H(I)$ is given by

$$H(I) = -P(I) \log P(I) - (1 - P(I)) \log(1 - P(I)), \quad (1)$$

where $P(I)$ is the probability that b_j is equal to 0. The binary entropy $H(I)$ is an approximation of the entropy according to Shannon's definition. Observing a deviation of the binary entropy is equivalent to observe a deviation of P. For non stego images, P follows a Gamma probability density function, whereas the probability density function is different for stego images. More surprisingly, P follows a normal $\mathcal{N}(0.5, \sigma)$ probability function and so, whatever the embedding rate, r, is, as shown in the figure 2. This difference of probability laws for stego and non stego images is explained by the avalanche criterion of the RLE and Huffman compression step. The avalanche criterion was introduced by Feistel [18] in 1973 for cryptographic purpose. It measures the number of bits of the cipher text which have been flipped when only one bit of the plain text has been flipped. Good cryptographic algorithm, and more particularly hash functions, are requeried to have an avalanche criterion close to 0.5. Let us denote Q the avalanche criterion of RLE and Huffman compression step, $P(I)$ the probability that b_j is equal to 0 before the embedding and $P'(I)$ the same probability after the embedding. Then $P'(I)$ is given by

$$P'(I) = P(I)(1 - Q) + (1 - P(I))Q. \quad (2)$$

If Q is close to 0.5 then $P'(I) \approx 0.5$. As shown in figure 3, when only few bits of the LSB of DCT coefficients are flipped, after RLE and Huffman compression almost half the bits are flipped. So, when embedding few bytes, $P'(I)$ becomes closer to 0.5. These phenomena is amplified since the avalanche criterion is close to 0.5 when only few bytes of DCT coefficients are changed and since steganography systems embed additional DCT coefficients to keep first order statistics unchanged. Whatever the amount of information we want to hide, $P'(I) \approx 0.5$ and thus makes possible the existence of steganalyzers which the detection rates are quasi-independent of the payload.
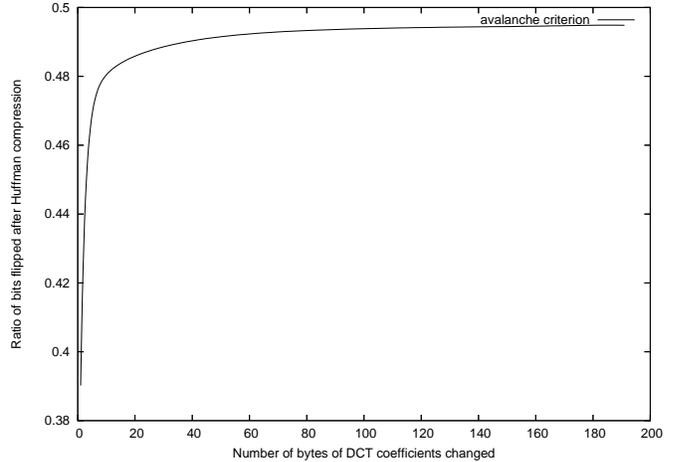
This approach gives us a new direction to look for. The main characteristics of RLE and Huffman compression step that make our steganalysis works in such a way are twice. First, this step can be viewed as a bijective function. Statistical deviations on the input variables will generate statistical deviations on the output variables, but looking for such deviations could be easier in the output domain and even be more discriminating. This is why using non linear Support Vector Machines are often more efficient than linear ones. Such functions can be compared to a magnifying glass that reveals and amplifies hidden statistical deviations. Then, the fact that this function has an avalanche criterion close to 0.5 implies that whatever the number of changes done on the input, half the bits of the output will be flipped. This conditions define a class of good functions to design stegnalysis schemes with efficiencies independent of the payload. RLE and Huffman is one of them.

We noticed a variation of higher order statistics of $\mathcal{C}_u((DCT_i), \mathcal{P}_2)$ when a message is embedded, despite $\mathcal{C}_u$ is a bijective mapping and the steganographic algorithm $\mathcal{A}$ tends to preserve the statistics of DCT coefficients. We have designed a steganographic distinguisher based on this deviations.

C. Universal Steganalysis Scheme

As previously, let I be a given JPEG image to analyse, (b_j) the output of $\mathcal{C}_u$ and $P(I)$ the probability that b_j is equal to 0. $P(I)$ can be seen as a global measurement. Now, let us see how JPEG steganography introduces local variations. To obtain a set of statistics on (b_j), we divide the stream (b_j) into blocks B_i of size s bytes, so that

$$B_i = b_{i \times 8s+1} \ldots b_{(i+1) \times 8s} \in \mathbb{F}_2^s. \quad (3)$$

We estimate the Hamming weights, $w(B_i) = \sum_{j=1}^{s} b_{i \times 8s+j}$, for the stream blocks. Naturally, variations of $P(I)$ will implies variations of $w(B_i)$ which can be seen as local mesurements.

Let $X \in \Omega = [0, 8s]$ be the random variable which values are the $w(B_i)$. We compute the probability density function
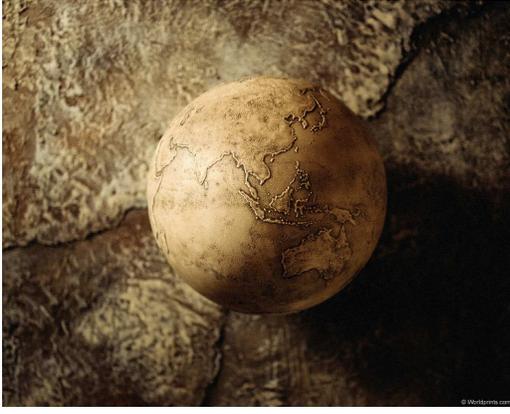
Figure 4. image_20099.jpg.

Figure 5. Deviation of weight for image_20099.jpg, using Outguess and $s = 8$.

followed by X and its moments of order i, $\mathcal{M}_i(I)$. As illustrated in figure 5, X does not follow the same probability density function whether I is a stego image or not. So, we experimentaly compute the average probability function $p(x)$ followed by X when I is a cover media (figure 5) and introduce the Kullbak-Leibler distance to measure the dissimilary between the observed probability density function $\hat{p}(x)$ and $p(x)$. $p(x)$ is computed taken randomly a huge amount of cover media and evaluating the density probability function follows by X for each image. $p(x)$ is then defined as the mean of all these functions. The Kulbak-Leibler distance $\mathcal{D}(\hat{p}, p)$ is defined by

$$\mathcal{D}(\hat{p}, p) = \sum_{x \in \Omega} \hat{p}(x) . \log \frac{\hat{p}(x)}{p(x)}. \tag{4}$$

As this distance is not symmetric, we introduce $\mathcal{D}_1(I) = \mathcal{D}(\hat{p}, p)$ and $\mathcal{D}_2(I) = \mathcal{D}(p, \hat{p})$. As illustrated in figure 5, the density probability function of X for the cover image of the figure 4 is very close to $p(x)$. On the contrary, we observed a deviation of this function after embedding the image with Outguess. We have noticed that embedding with Outguess, F5 and JPHide and JPSeek increases the Kulbak-Leibler distance between the observed density probability function of X and $p(x)$.

Now, we will map I to the statistical vector $\mathcal{V}(I)$ of $k + 4$ coordinates defined by

$$I \longrightarrow \mathcal{V}(I) = (\mathcal{M}_0(I), \dots, \mathcal{M}_k(I), P(I), \mathcal{D}_1(I), \mathcal{D}_2(I)), \tag{5}$$

and design an universal steganalysis scheme which the parameters are (s, k). Each component of the statistical vector does not follow the same probability density function whether the image I is a stego one or not.

We have pointed out some statistics in the compressed frequency domain which are liable to deviation when information is embedded with a steganographic algorithm such as Outguess, F5 or JPHide. These statistical features are independent of a specific algorithm, but to prove our scheme is an efficient universal steganalysis scheme we still have to evaluate its universality property, as discussed in the introduction.
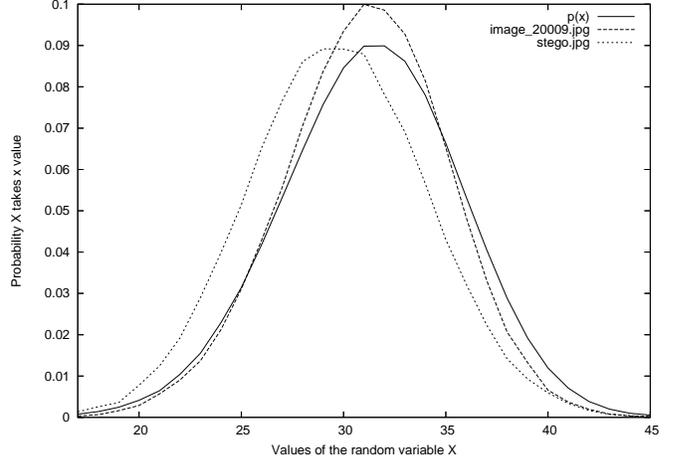
IV. EXPERIMENTAL RESULTS

A. Classifier Design

We need a set, $\mathcal{C}$ of cover media and a set, $\mathcal{S}$ of stego images. For convenience, these samples have the same cardinality n, but the following method can be easily adapted with learning sets of different cardinalities.

First, for each set, we compute $\mathcal{V}_c = \{\mathcal{V}(I) | I \in \mathcal{C}\}$ as defined in (5), and $\mathcal{V}_s = \{\mathcal{V}(I) | I \in \mathcal{S}\}$ which are subsets of $\mathbb{R}^{k+4}$. We denote g_c, respectively g_s, the barycenter of $\mathcal{V}_c$, respectively $\mathcal{V}_s$, and g the barycenter of g_c, g_s. Then, we take g as the origin of the system of coordinates and compute the covariance matrices, V_c and V_s. Finally, we compute the intraclass and interclass variance matrices W and B defined under our hypothesis by

$$B = \frac{1}{2}(g_c - g_s)(g_c - g_s)', \tag{6}$$

$$W = \frac{1}{2}(V_c + V_s). \tag{7}$$

The variance matrix, V is given by $V = B + W$.

The Fisher discrimination analysis [19] consists in finding a projection axis which discriminates the best $\mathcal{V}_c$ and $\mathcal{V}_s$ and thus $\mathcal{C}$ and $\mathcal{S}$. This axis, (g_c, g_s), is defined by the vector

$$u = W^{-1}(g_c - g_s), \tag{8}$$

where $M = W^{-1}$ can be considered as a metric. Actually, a new image I, represented by the point p will be said to be in $\mathcal{C}$, if $d^2(p, g_c) > d^2(p, g_s)$, where d is a distance based on the metric M. According to the Mahalanobis-Fisher rule, we decide that I is in $\mathcal{C}$ if and only if

$$p.u = pM(g_c - g_s) > T, \tag{9}$$

where T is the detection threshold. Another metric can also be considered, setting $M = V^{-1}$.

B. Coding Considerations

In this subsection, we summarize formally the methodology we followed to designed our classifier. As classifying techniques our steganalysis scheme is divided into two main parts, which are the *learning step* and then *the detection step*. The first one is time consuming but can be done off-line, only once. The learning step provides a set of parameters for the detection algorithm. The detection algorithm is very quick and can be run on-line. Moreover, the same parameters can be used for all the images to be analysed.

Learning Algorithm

Input: $\mathcal{A}'$ a set of steganography algorithms.
$\mathcal{C}$ a set of size n of cover-media.
$\mathcal{S}$ a set of size n stego-media embedded with algorithms in $\mathcal{A}'$.

Parameters: k the greatest order of moments of the variable X.
s the size of the blocks.

Output: u, a vector of $\mathbb{R}^{k+4}$.
T, a detection threshold.

1) Compute $\mathcal{V}_c = \{\mathcal{V}(I) | I \in \mathcal{C}\}$, according to equation 5.
2) Compute $\mathcal{V}_s = \{\mathcal{V}(I) | I \in \mathcal{S}\}$, according to equation 5.
3) Compute (g_c, g_s) the barycenters of $\mathcal{V}_c$ and $\mathcal{V}_s$ such as

$$(g_c, g_s) = \frac{1}{n} \left(\sum_{I \in \mathcal{C}} \mathcal{V}(I), \sum_{I \in \mathcal{S}} \mathcal{V}(I) \right).$$

4) Compute W^{-1} and V^{-1} according to equations 6 and 7.
5) Choose $M \in \{W^{-1}, V^{-1}\}$ and $T \in \mathbb{R}$ such as (M, T) equals

$$Argmax \left(\begin{array}{c} \#\{I \in \mathcal{C} | \mathcal{V}(I).M.(g_c - g_s) \geq T\} + \\ \#\{I \in \mathcal{S} | \mathcal{V}(I).M.(g_c - g_s) < T\} \end{array} \right).$$

6) **Return** $u = M.(g_c - g_s)$, and T.

At that point, we first have to do some compromises. On one hand, the detection algorithm will be all the more accurate as the training sets are bigger. On the other hand, the coefficients of W and V are real numbers and then we can not do some approximations during the computation of W^{-1} and V^{-1}. Otherwise, the result of the equation 8 may be far from the theoretic value of u and the detection algorithm may diverge. To code this algorithm, we decided to use the Gnu Multiple Precision Arithmetic Library and encode all the coefficients with exact rational numbers. Trivialy, arithmetic operations with such numbers are no more linear in the size of the operands. All the more, the size of these numbers increases with the cardinality of the training sets, n.

Detection Algorithm

Input: I a JPEG image to analyse.

Parameters: (u, T) provided by the Learning Algorithm.
k the greatest order of moments of the variable X.
s the size of the blocks.

Output: *"Stego-medium"* or *"cover-medium"*

1) Compute $\mathcal{V}(I)$ according to equation 5.
2) **if** $\mathcal{V}(I).u > T$ **then return** "cover-medium"
   else return "stego-medium".

The time and space complexities are trivialy linear in the size of the analysed JPEG file. This algorithm can be efficiently coded as all the statistical features are based on counting the number of 0-bits in the data stream.

C. Learning Step

For each training of our classifiers, we used between 3,000 and 4,000 randomly choosen images from a database of about 100,000 JPEG images downloaded from the web, notably *https://www.worldprints.com* in 2000. The database is composed of grayscale and color images of different sizes. We disposed of a set $\mathcal{A} = \{Outguess, F5, JPHide\}$ of three known algorithms, for which known specific attacks exist. Our goal was to produce a subset $A' \subset \mathcal{A}$ for training our classifier so as to, at least the use of one algorithm in $\bar{A}'$ can be efficiently detected. We chose to configure our scheme with $k = 3$, which is a good trade off between reasonable computing time and a good detection accuracy. We tested different values for s: 8, 16, 32 and 64. To show the effectiveness of our approach with very low embedding rates, we mixed stego images with an embedding rate from 10^{-6} to 10^{-2}.

We tried all the subsets but $\mathcal{A}$ and the empty set. For illustration, we give in table I the best experimental parameters obtained for the subsets $A_1 = \{F5, JPHide\}$ and $A_2 = \{Outguess, JPHide\}$, of two algorithms. The training set for A_1 was composed of 2,000 cover media and 2,000 stego images embedded respectively by F5 and JPHide with the embedding rates 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3} and 10^{-2}. The training set for A_2, figure 6, was composed of 1,500 cover media and 1,500 stego images embedded by Outguess and JPHide with the same embedding rates. For each training set, we determined the discriminant factor u for the metrics W^{-1} and V^{-1} as defined in section IV-A. We also chose the value for s which gave us the best detection rate for the training set, in order to preserve the detection of cover media during the wild detection step.

The main reason we chose the parameter k less than 4 in practice, is motivated by the fact that $\hat{p}(x)$ seems to be $p(x)$
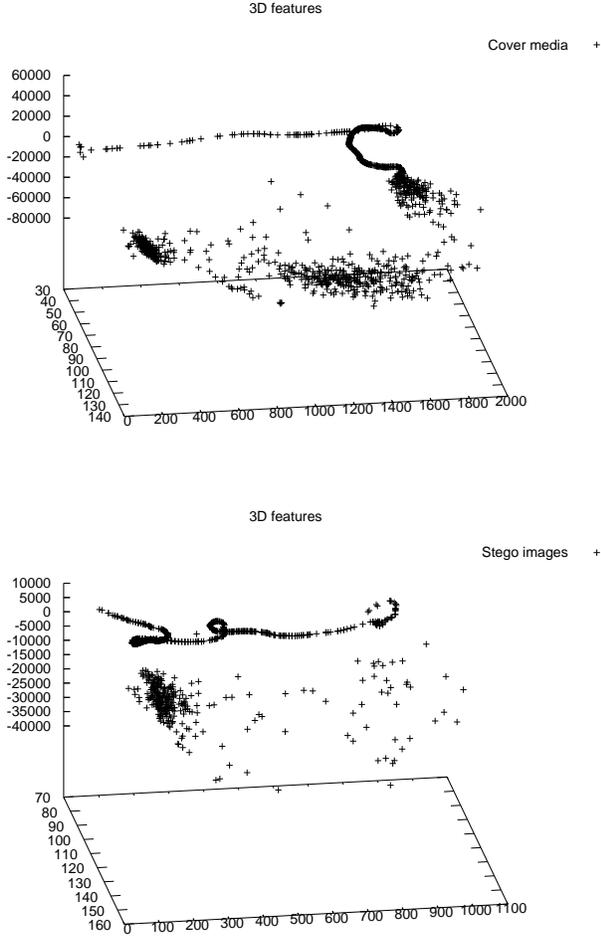
Figure 6. Statistical vectors for A_2 projected onto $\mathbb{R}^3$. On the top $\mathcal{V}_c$, on the bottom $\mathcal{V}_s$.

TABLE I.
OPTIMAL PARAMETERS FOR A_1 AND A_2.

	A_1	A_2
Metric	W^{-1}	W^{-1}
threshold	-0.2576	-0.2272
s	16	16
u	$\begin{pmatrix} -2.786796E-02 \\ -7.513114E-04 \\ +4.230118E-05 \\ +7.628112E-07 \\ +8.246617E-02 \\ -1.217546E+00 \\ +6.209087E-02 \end{pmatrix}$	$\begin{pmatrix} -1.479200E-02 \\ -3.510628E-04 \\ +2.590442E-05 \\ +4.470248E-07 \\ +9.995430E-02 \\ -1.250042E+00 \\ +3.160536E-01 \end{pmatrix}$

shifted to the left as illustrated in figure 5. But, the global shape of $\hat{p}(x)$ is close to this of $p(x)$, that implies that the random variables they represent will only differ in a significant way for the lowest orders of statistical moments. That means, moments of order higher than 4 will not be discriminating. Moreover, as we explained in section IV-B, computing W^{-1} and V^{-1} is very time expensive since their coefficients are exact rational numbers. As evaluating W^{-1} and V^{-1} costs $\mathcal{O}(k^3)$ arithmetic operations, constraining k to the really

discriminating moments may save some computational time.

D. Wild Detection Step

To show the efficiency of our scheme, we randomly generated challenge sets composed of 1,000 cover media and 1,000 stego images embedded with an embedding rate from 10^{-6} to 10^{-2}. After having trained two distinguishers D_1 and D_2, as explained in section IV-C, with respectively A_1 and A_2, we made them detect the use of new algorithms, Outguess for D_1 and $F5$ for D_2. The efficiency of D_1 and D_2 is presented in the figure 7. Two main conclusions can be drawn when observing these results. Firstly, D_1 and D_2 are able to detect efficiently the use of an algorithm which has not been used in the training set, that proves the universality property of our scheme. Nevertheless, this only proves that our detectors may detect images embed with an unknown steganographic algorithm but with no confidence it will work with all unknown steganographic algorithm. Finally, the detection rate appears to be constant and independent of the embedding rate (figures 7 and 8), according to the hypothesis we made in section III-B.
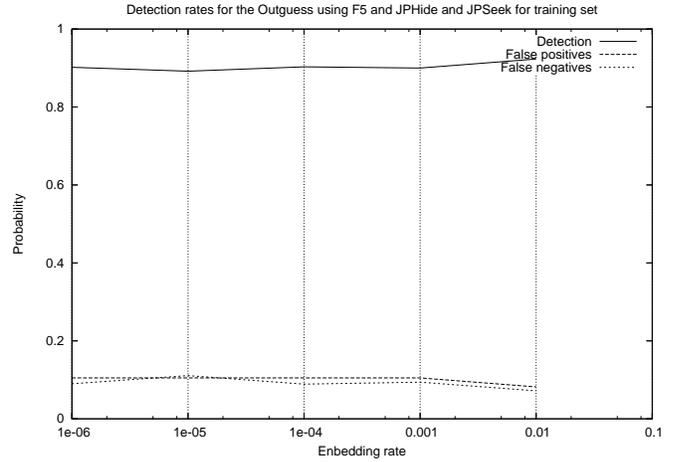

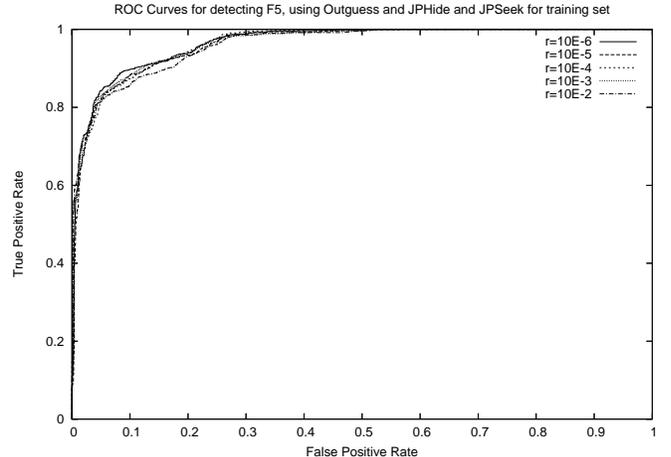Figure 7. Detection curves for D_1 detecting Outguess.

Figure 8. ROC curves for D_2 detecting F5 for different embedding rates, r.

More precisely, the characteristics of D_1 and D_2 are summarized in table II. We observed what follows:

- D_1 detects the use of Outguess with detection rate 90,47%, positive error rate 10% and negative error rate 9,12%.
- D_2 detect the use of F5 with detection rate 88,97%, positive error rate 9,56% and negative error rate 12,54%.

TABLE II.
CHARACTERITICS OF D_1 AND D_2.

	D_1	D_2
Detected algorithm	Outguess	F5
False positive rate	10%	9,56%
Recall	90,93%	87,47%
Precision	90%	90,17%
F-Score	81,92%	78,87%
Accuracy	90,47%	88,97%

Obviously, these results depend on the distribution of cover media and stego images, but they give us a lower bound of the dectection rate. For both, the worste cases are obtained with sets only composed of cover media. So, for D_1, the detection rate is higher than 90% and for D_2 higher than 87%, whatever the distribution of cover media and stego images is. More results obtained detecting the Outguess using a subset of $\mathcal{A}$ composed of only one algorithm, F5, can be found in [4].

V. CONCLUSION

We have proposed a new approach for universal JPEG steganalysis which is based on statistics of the compressed frequency domain and benefits from the statistical deviation of the entropy of the binary output stream. Exploring the compressed frequency domain completes the traditional detection schemes and reveals a new class of good functions for steganalysis. These functions are requiered to be bijective and have an avalanche criterion close to 0.5. Under this hypothesis, designing steganography classifiers which the accuracies do not depend on the payload may be possible. If the function is well choosen it could reveal and magnify statistical deviations which are not visible in its input domain. Looking for such functions appears to be promising. Not only a theoretical view, one of them, defined by the RLE and Huffman compression, has been evaluated. The avalanche criterion of the JPEG lossless compression step makes this deviation quasi-independent of the embedding rate and so, makes possible the design of steganographic detectors which the efficiencies do not depend on the payload. We design such a steganalyzer with very high and constant detection rates, as illustrated in section IV-D. The experimental results show that our steganalysis scheme is able to efficiently detect the use of new algorithms which are not used in the training step, even if the embedding rate is very low ($\approx 10^{-6}$).

Since universal detectors are less accurate but more general than specific ones, they are more oriented to detect the use of unknown steganographic systems or those for which no specific attack is known. Universal steganalysis provides another kind of detection services and should be run in parallel with specific detectors, for instance in a operational steganalytic system.

In future researches, we will try to improve the efficiency of our scheme using Support Vector Machines instead of Fisher discriminant. We hope to benefit from the non linearity of certain kernels and so increase our detection rates. This new universal scheme pointes out new statistics features we will combine to improve our specific steganalytic techniques. We are also working on combining detectors running in different domains.

REFERENCES

[1] N. Provos, "Defending against statistical steganalysis." in *10th USENIX Security Symposium*, 2001.

[2] A. Westfeld, "F5-a steganographic algorithm." in *Information Hiding*, ser. Lecture Notes in Computer Science, I. S. Moskowitz, Ed., vol. 2137. Springer, 2001, pp. 289–302.

[3] A. Latham, "Steganography: JPHIDE AND JPSEEK," 1999, http://linux01.gwdg.de/~alatham/stego.html.

[4] J. Barbier, E. Filiol, and K. Mayoura, "Universal jpeg steganalysis in the compressed frequency domain." in *IWDW*, ser. Lecture Notes in Computer Science, Y.-Q. Shi and B. Jeon, Eds., vol. 4283. Springer, 2006, pp. 253–267.

[5] I. Avicibaş, N. Memon, and B. Sankur, "Steganalysis based on image quality metrics." in *Proc. SPIE, Security and Watermarking of Multimedia Contents III*, vol. 4314, 2001, pp. 523–531.

[6] H. Farid, "Detecting hidden messages using higher-order statistical models." in *ICIP (2)*, 2002, pp. 905–908.

[7] S. Lyu and H. Farid, "Detecting hidden messages using higher-order statistics and support vector machines." in *Information Hiding*, ser. Lecture Notes in Computer Science, F. A. P. Petitcolas, Ed., vol. 2578. Springer, 2002, pp. 340–354.

[8] Lyu and Farid, "Steganalysis using color wavelet statistics and one-class support vector machines." in *Proc. SPIE, Security and Watermarking of Multimedia Contents VI*, 2004.

[9] S. Lyu and H. Farid, "Steganalysis using higher-order image statistics." *IEEE Transactions on Information Forensics and Security*, no. 1, 2006.

[10] A. Westfeld and A. Pfitzmann, "Attacks on steganographic systems." in *Information Hiding*, ser. Lecture Notes in Computer Science, A. Pfitzmann, Ed., vol. 1768. Springer, 1999, pp. 61–76.

[11] J. J. Fridrich, M. Goljan, and D. Hogea, "Steganalysis of jpeg images: Breaking the f5 algorithm." in *Information Hiding*, ser. Lecture Notes in Computer Science, F. A. P. Petitcolas, Ed., vol. 2578. Springer, 2002, pp. 310–323.

[12] J. Fridrich, M. Goljan, and D. Hogea, "New methodology for breaking steganographic techniques for jpegs." in *Proc. SPIE, Security and Watermarking of Multimedia Contents V*, January 2003, pp. 143–155.

[13] J. J. Fridrich, "Feature-based steganalysis for jpeg images and its implications for future design of steganographic schemes." in *Information Hiding*, ser. Lecture Notes in Computer Science, J. J. Fridrich, Ed., vol. 3200. Springer, 2004, pp. 67–81.

[14] J. Fridrich and T. Pevny, "Multiclass blind steganalysis for jpeg images." in *Proc. SPIE, Security and Watermarking of Multimedia Contents VIII*, January 2006.

[15] J. J. Harmsen and W. A. Pearlman, "Kernel fisher discriminant for steganalysis of jpeg hiding methods." in *ACM Multimedia and Security*, August 1-2 2005.

[16] G.-S. Lin, C. H. Yeh, and C. C. J. Kuo, "Data hiding domain classification for blind image steganalysis." in *ICME*, 2004, pp. 907–910.

[17] J. Barbier, E. Filiol, and K. Mayoura, "New features for specific jpeg steganalysis." in *Enformatika*, ser. Transactions on Engineering, Computing and Technology, C. Ardil, Ed., vol. 16. World Enformatika Society, 2006, pp. 72–77.

[18] H. Feistel, "Cryptography and computer privacy." *Scientific American*, vol. 228, no. 5, pp. 15–23, 1973.

[19] G. Saporta, *Probabilité, analyse des données et statistiques (in french)*. Technip, 1990.

[20] Y.-Q. Shi and B. Jeon, Eds., *Digital Watermarking, 5th International Workshop, IWDW 2006, Jeju Island, Korea, November 8-10, 2006, Proceedings*, ser. Lecture Notes in Computer Science, vol. 4283. Springer, 2006.

[21] C. Ardil, Ed., *3rd International Conference on Computer, Information, and Systems Science, and Engineering, CISE 2006, Venice, Italy, November 24-26, 2006, Proceedings*, ser. Transactions on Engineering, Computing and Technology, vol. 16. World Enformatika Society, 2006.

[22] C. W. Brown and B. J. Shepherd, *Graphics File Formats, reference and guide*. Manning, 1995.

[23] T. Kalker, I. J. Cox, and Y. M. Ro, Eds., *Digital Watermarking, Second International Workshop, IWDW 2003, Seoul, Korea, October 20-22, 2003, Revised Papers*, ser. Lecture Notes in Computer Science, vol. 2939. Springer, 2004.

[24] R. Chandramouli, M. Kharrazi, and N. D. Memon, "Image steganography and steganalysis: Concepts and practice." in *IWDW*, ser. Lecture Notes in Computer Science, T. Kalker, I. J. Cox, and Y. M. Ro, Eds., vol. 2939. Springer, 2003, pp. 35–49.

[25] F. A. P. Petitcolas, Ed., *Information Hiding, 5th International Workshop, IH 2002, Noordwijkerhout, The Netherlands, October 7-9, 2002, Revised Papers*, ser. Lecture Notes in Computer Science, vol. 2578. Springer, 2003.

[26] I. S. Moskowitz, Ed., *Information Hiding, 4th International Workshop, IHW 2001, Pittsburgh, PA, USA, April 25-27, 2001, Proceedings*, ser. Lecture Notes in Computer Science, vol. 2137. Springer, 2001.

[27] J. J. Fridrich, Ed., *Information Hiding, 6th International Workshop, IH 2004, Toronto, Canada, May 23-25, 2004, Revised Selected Papers*, ser. Lecture Notes in Computer Science, vol. 3200. Springer, 2004.

[28] G. J. Simmons, "The prisoners' problem and the subliminal channel." in *CRYPTO*, 1983, pp. 51–67.

[29] G. K. Wallace, "The jpeg still picture compression standard," *Commun. ACM*, vol. 34, no. 4, pp. 30–44, 1991.

[30] A. Pfitzmann, Ed., *Information Hiding, Third International Workshop, IH'99, Dresden, Germany, September 29 - October 1, 1999, Proceedings*, ser. Lecture Notes in Computer Science, vol. 1768. Springer, 2000.

Johann Barbier is currently a PhD candidate at École Polytechnique, Palaiseau, France, with the Virology and Cryptology Lab at the French Army Signals Academy, Rennes, France. He received his MS degree in algorithmic in 2003 from École Polytechnique. He was graduate from the French Military Academy of Saint-Cyr, Coëtquidan, France in 2002 and received a MS degree in computer science engineering.

He is currently a Cryptologist Engineer at the Cryptology Department of the Center for Armament Electronics, Bruz, France. He is a lecturer at the French Military Academy of Saint-Cyr since 2002 and at the French Army Signals Academy since 2003. His research interests include symmetric cryptology, steganography and error correcting codes reconstruction.

Kichenakoumar Mayoura is currently a PhD student with the Virology and Cryptology Lab at the French Army Signals Academy, Rennes, France. He received his MS degree in mathematics in 2004 at the University of Le Mans, France.

He is a lecturer at the French Army Signals Academy. His research interests include steganography and cryptology.

Éric Filiol received his PhD degree in computer science from École Polytechnique, Palaiseau, France, in 2001. He received his MS degree in algorithmic in 1997 from École Polytechnique.

Lieutenant-Colonel Filiol is currently the Head scientist officer of the Virology and Cryptology Lab at the French Army Signals Academy, Rennes, France. He is professor at the French Army Signals Academy and lecturer at many Universities. His research interests include cryptanalysis techniques of symmetric cryptosystems, combinatorics in cryptography and virology.

Prof. Filiol is a recipient of the French award of book and communication in technology, the Roberval award, in 2004 for his book *Computer viruses: from theory to applications*. He is also the Editor-in-Chief of the *Journal in Computer Virology*, Springer.