New features for specific JPEG Steganalysis

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

[1] École Supérieure et d'Application des Transmissions,
Laboratoire de Virologie et Cryptologie,
BP 18, 35998 Rennes Cedex, France
[2] Centre d'Électronique de l'ARmement, Département de Cryptologie,
La Roche Marguerite, BP 57419,
35174 Bruz Cedex, France,
`johann.barbier@dga.defense.gouv.fr`

Abstract. We present in this paper a new approach for specific JPEG steganalysis and propose studying statistics of the compressed DCT coefficients. Traditionally, steganographic algorithms try to preserve statistics of the DCT and of the spatial domain, but they cannot preserve both and also control the alteration of the compressed data. We have noticed a deviation of the entropy of the compressed data after a first embedding. This deviation is greater when the image is a cover medium than when the image is a stego image. To observe this deviation, we pointed out new statistic features and combined them with the Multiple Embedding Method. This approach is motivated by the *Avalanche Criterion* of the JPEG lossless compression step. This criterion makes possible the design of detectors whose detection rates are independent of the payload. Finally, we designed a Fisher discriminant based classifier for well known steganographic algorithms, Outguess, F5 and Hide and Seek. The experiemental results we obtained show the efficiency of our classifier for these algorithms. Moreover, it is also designed to work with low embedding rates ($< 10^{-5}$) and according to the avalanche criterion of RLE and Huffman compression step, its efficiency is independent of the quantity of hidden information.

Keywords: steganalysis, JPEG, Fisher discriminant, avalanche criterion.

Introduction

Steganogaphy is an old science which takes its roots in antique Greece. Litteraly, steganography means *"art of covered writing"*. For a long, steganography was rudimentary and its use was exclusively reserved to the military and secret services. But the communication society enables acces to a mass of numeric information. This huge amount of information alows to hide easily some messages and to communicate in a discreet way. With the invention of the internet, lots of steganographic softwares have been developped for many numeric covers like images, mp3, filesystems, texts, emails etc... Naturaly, most of them are dedicated to JPEG standard for it is one of the spreadest formats to store and exchange images. Modern steganography takes its origins in 1983 with the paper of G. Simmons, *"Prisoner' problem and the subliminal channel"* [18], and a new active research branch dealing with steganography appeared about ten years ago.

The context is the following one. Alice and Bob are in jail and want to plan their escape. Their only way to communicate is Wendy, the warden. Wendy stops delivering messages as soon as she can prove a message contains information for an escape plan. For confidentiality of communications, cryptography is entirely well adapted. But, in our case, if the message is ciphered, Wendy can force Alice or Bob to decipher it with their own key and then prove they are plotting to escape. The steganography provides Alice and Bob the way to communicate discreetly and so a new security service in addition to confidentiality: *plausible deniability*. Now, Wendy has first to detect if hidden information is embedded in the message and then retrieve these information to prove its existence. Alice and Bob could always hide an innocuous message in addition to their plans, and so reveal only the former if they are forced to.

To achieve this, Alice and Bob need first to agree on a compression algorithm $\mathcal{C}$, a randomized steganographic algorithm $\mathcal{S}$, and a secret key K. We also suppose that Alice and Bob have the ability to generate their own set of cover media $C_A = \{C_A^i\}$ and $C_B = \{C_B^i\}$. Each has only access to its own set of cover media. Alice wants to send a message M to Bob trough Wendy. First, she compresses M to $M^{'} = \mathcal{C}(M)$ to reduce the message length and make it seemed like random, in order to minimize the number of changes in the cover medium. Then, she chooses one cover medium, C_A^i, and embeds it with $M^{'}$ and $\mathcal{S}$ to obtain $C_A^{i'} = \mathcal{S}(K, M^{'}, C_A^i)$. To retrieve M, Bob computes $M^{'} = \mathcal{S}^{-1}(K, C_A^{i'})$ and $M = \mathcal{C}^{-1}(M^{'})$.

In this paper, we take place in Wendy's shoes, and our goal is to detect the existence of embedded message into JPEG images. According to Kerchoffs' principles, $\mathcal{S}$ and $\mathcal{C}$ are known and only K is kept secret. Our new approach is illustrated with the well known steganographic algorithms, Outguess [16], F5 [20] and JPHide [10]. In the same way, it can also be adapted to detect the use of another algorithms. 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 statistic features to detect hidden information in JPEG images. For each steganographic algorithm examined, these features do not follow the same propability density function whether JPEG image is embedded or not.

In the first section, we quickly present the JPEG standard and DCT-based steganography. We also present a new approach for JPEG steganalysis and define the statistic features we will use to detect steganographic contents. In the second section, we recall state of the art JPEG steganalysis techniques, discuss of *specific* versus *universal* steganalysis and put our approach back in its place. Then, we present the Multiple Embedding Method and its application to Outguess, F5, and JPHide algorithms. In section 3, we explain the design of our Fisher classifier and detail the experimental framework and the results we obtained. Finally, we conclude in the last section and give some discussions.

1 JPEG Steganography

1.1 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[3] recommendation T.81 and the ISO[4] 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[5]-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.

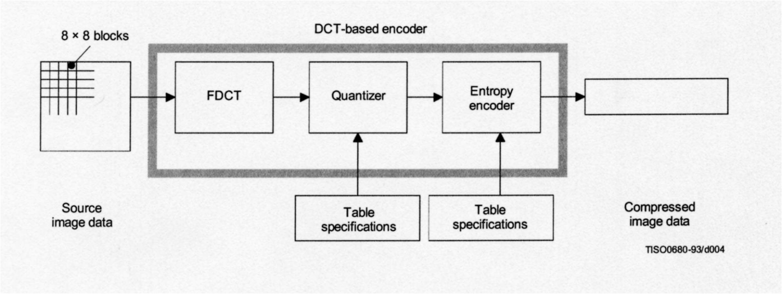

Fig. 1. DCT-based encoder simplified diagram

[3] International Telegraph and Telephone Consultative Committee

[4] International Standard Organisation

[5] Discrete Cosine Transform

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[6] is used to compress the high frequency coefficients. Then a Huffman coding procedure is applied. Finally, the output is the JPEG raw binary data.

1.2 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 intuitively 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 compressions. So, the only practical way to embed an information is in DCT coefficients, after RLE or Huffmann compressions. To minimize the distortions

[6] Run Length Encoding

of the original image, DCT are the most adapted.

The main problem, when embedding information in DCT coefficients, is to preserve the statistics of the cover medium. Most of new steganographic systems take care of keeping DCT statistics unchanged, histogram for example, but even if DCT statistics are preserved, many steganalysis [1, 4, 12–14] are based on deviations of some decompressed cover image statistics. It seems that both cannot be preserved at the same time.

2 Detecting JPEG stego images

2.1 JPEG steganalysis methods

Different approaches have been used to detect stego images. The first one consists in studying directly DCT coefficients like J. Fridich [5, 6] 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 [7, 8] 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 [4, 12] or with wavelet transform statistics [13, 14] of decompressed JPEG image. J. J. Harmsen et al. [9] proposed to use a Fisher discriminant instead of a SVM and I. Avicib et al. [1] introduced metrics based on images quality.

Previous methods have even been used together [11] to increase the accuracy of detectors. Among these techniques we can distinguish two categories of steganalysis: *specific steganalysis* and *universal steganalysis.*

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 developped in [5–7, 9] are specific.

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 [1, 4, 8, 11–14] are universal.

In this paper, we will study a specific method adapted for the compressed frequency domain. This technique can be adapted to detect the use of many JPEG steganographic algorithms.

2.2 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 $\mathcal{C}_u((DCT_i), \mathcal{P}_2)$.

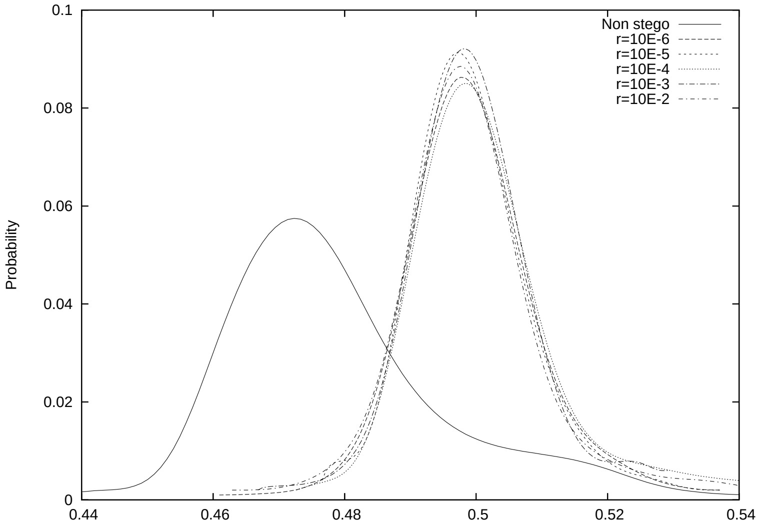

Fig. 2. Density probability functions of P for JPHide stego and non stego images.

Let I a given JPEG image to analyse and (b_j)[7] the output of $\mathcal{C}_u$. We noticed a variation of the entropy of the output stream when the image has been embedded with a steganographic scheme. The binary entropy $H(I)$ is given by

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

where $P(I)$ is the probability that b_j is equal to 0. 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

[7] (b_j) is only composed of the RLE and Huffman compressed DCT coefficients and does not include the JPEG file header.

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 [15] of the RLE and Huffman compression step. As shown in figure 3, when only few bits of the DCT coefficients LSB 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. This criterion makes possible the existence of steganalysers which the detection rates are quasi-independent of the payload.

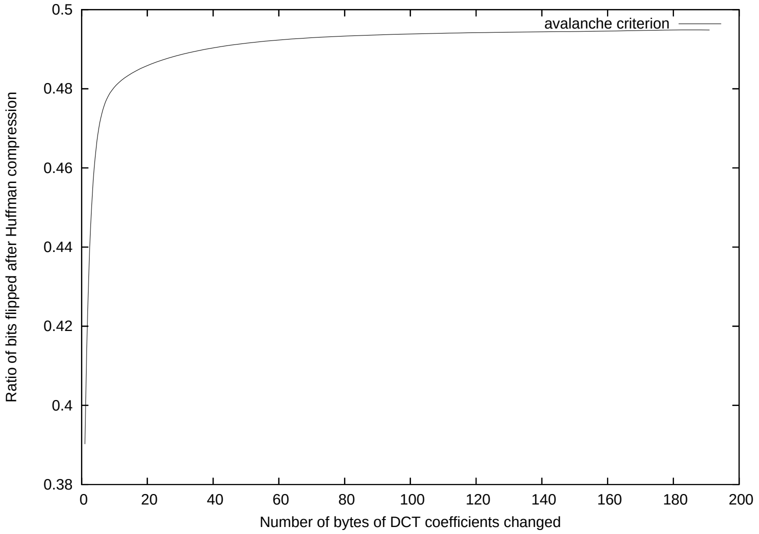

Fig. 3. Avalanche criterion of RLE+Huffman compression function.

Because of the entropy deviation, we compute for a given image I the average number of bits $\mathcal{M}(I)$ which the value egals 0, where

$$\mathcal{M}(I) = \frac{1}{m}\sum_{j=1}^{m}(1-b_j). \tag{2}$$

For non stego images, $\mathcal{M}$ can be seen has a random variable which follows a Gamma density probability function. For stego images, $\mathcal{M}$ follows a $\mathcal{N}(0.5,\sigma)$ density probabilty function as illustrated in figure 2.

2.3 The Multiple Embedding Method

To describe the Multiple Embedding Method (MEM), we first need a steganographic algorithm $\mathcal{S}$, a JPEG image I, the size of the stego key k, and the length of the message to embed, l. We will also denote the relative message length $\rho = \frac{l}{|I|}$ where $|I|$ is the size of I. ρ is also called the embedding rate. n stego keys K_i and messages M_i of length l, $i = 1 \dots n$ are randomly generated.

Now, let us denote the sequence $\mathcal{I} = (I_i)_{i=0 \dots n}$ defined by

$$\begin{cases} I_0 = I, \\ I_i = \mathcal{S}(K_i, M_i, I_{i-1}), \forall i = 1 \dots n. \end{cases} \tag{3}$$

To process the variation of $\mathcal{M}$, we compute the sequence $\varDelta = (\varDelta_i)_{i=0 \dots n}$ defined by

$$\begin{cases} \varDelta_0 = 0, \\ \varDelta_i = |\mathcal{M}(I_i) - \mathcal{M}(I_{i-1})|, \forall i = 1 \dots n. \end{cases} \tag{4}$$

We have noticed that if I hasn't been embedded by $\mathcal{S}$, then we have

$$\begin{cases} \varDelta_1 \gg \varDelta_i, \ \ \forall i > 1, \\ \varDelta_i \text{ and } \varDelta_j \text{ are of the same order of magnitude}, \forall i, j > 1, \end{cases} \tag{5}$$

and $\varDelta_i$ and $\varDelta_j$ are of the same order of magnitude $\forall i, j$, otherwise. To catch this fact, we also define the sequence $Q = (Q_i)_{i=0 \dots n}$, by

$$\begin{cases} Q_i = 0, & \forall i = 0 \dots 1, \\ Q_i = \frac{\varDelta_i}{\varDelta_{i-1}} \text{ when defined, } \infty \text{ otherwise}, & \forall i = 2 \dots n. \end{cases} \tag{6}$$

The equation (5) implies

$$Q_2 \gg 1 \tag{7}$$

if I has not already been embedded and $Q_2 \approx 1$, otherwise. With these sequences, we are now able to build a naive steganalytic scheme for $\mathcal{S}$ as follows.

Multiple Embedding Method :
Input : a JPEG image I, k the size of the random stego-keys K_i and l the size of the random message M_i.
Ouput : *"S-stego image"* or *"non S-stego image"*.

1. compute the sequence $\mathcal{I}$ with $I_0 = I$,
2. compute the sequences $\varDelta$ and Q,
3. if (5) and (7) hold then return *"non S-stego image"*,
4. return *"S-stego image"* otherwise.

Fig. 4. image_04173.jpg.

We computed the previous sequences for the non $\mathcal{S}$-stego image *image_04137.jpg*, figure 4, with the following parameters: $|I| = 413830$ bytes, $l = 1$ byte, $\rho = 2.42\ 10^{-6}$. We obtained for Outguess, F5 and JPHide the results descriped in table 1. It is easy to see that (5) and (7) hold. Now, by definition, I_1 is a S-stego image and its sequence of MEM statistics can be read from table 1 by shifting upward the rows and setting the Δ_0 and the Q_1 to 0. In that case, as claimed previously, (5) and (7) do not hold. This instance also shows that a $\mathcal{S}$-stego JPEG image with only few bytes embedded, can be detected with MEM.

	Outguess			F5			JPHide		
i	$\mathcal{M}_i$	Δ_i	Q_i	$\mathcal{M}_i$	Δ_i	Q_i	$\mathcal{M}_i$	Δ_i	Q_i
0	0.5119	0	0	0.5119	0	0	0.5119	0	0
1	0.5287	1.676E-2	0	0.5137	1.799E-3	0	0.5017	1.016E-2	0
2	0.5287	1.958E-6	8563	0.5137	1.311E-5	137	0.5018	1.227E-5	828
3	0.5287	2.181E-5	8.972E-2	0.5137	3.302E-6	3.968	0.5018	1.159E-5	1.058
4	0.5287	2.147E-5	1.016	0.5137	8.528E-5	3.870E-2	0.5018	3.907E-5	0.297

Table 1. Statistics sequences of *image_04173.jpg* for Outguess, F5 and JPHide.

Remark: the choice of the parameters l and k is not significant. Actually, the size of the stego-keys does not have any impact on the amount of DCT coefficients changed. Moreover, l does not change the accuracy of detecting the variation of $\mathcal{M}$ since this variation has shown to be quasi-independent of the embedding rate, in section 2.2.

2.4 Detecting the Outguess, F5, and JPHide

To improve this technique we also benefit from the different probability density functions followed by $\mathcal{M}(I)$ when I is a non $\mathcal{S}$-stego image and when I is a $\mathcal{S}$-stego image. So, for a given JPEG image I, we compute $\mathcal{M}_I = \mathcal{M}(I)$, $\Delta_I = \Delta_1$

and $Q_I = Q_2$ and map I to the statistic vector $\mathcal{V}(I)$ defined by

$$I \longrightarrow \mathcal{V}(I) = (\mathcal{M}_I, \varDelta_I, Q_I). \tag{8}$$

Each component of $\mathcal{V}(I)$ does not follow the same probabiblity density function whether I is a $\mathcal{S}$-stego image or not. We will now underline these different probability density functions for the mean $\mathcal{M}$, the delta $\varDelta$, and the ratio Q, for Outguess, F5 and JPHide.

Outguess The Outguess steganographic algorithm [16] 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 reduntant 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.

F5 The F5 steganographic algorithm [20] 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 coefficents. 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 preserves the DCT histogram 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.

JPHide JPHide is a steganographic system developped by A. Latham in 1999 [10] which embeds data in LSB of the DCT coefficients. It uses a PRNG based on Blowfish.

3 Experimental results

3.1 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 cardinals.

First, for each set, we compute $\mathcal{V}_c = \{\mathcal{V}(I) | I \in \mathcal{C}\}$ as defined in (8), and $\mathcal{V}_s = \{\mathcal{V}(I) | I \in \mathcal{S}\}$ which are subsets of $\mathbb{R}^3$. 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

$$B = \frac{1}{2}(g_c - g_s)(g_c - g_s)^{'}, \tag{9}$$

$$W = \frac{1}{2}\left(V_c + V_s\right). \tag{10}$$

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

The Fisher discrimination analysis [17] consists in finding a projection axis which discriminates the best $\mathcal{V}_c$ and $\mathcal{V}_s$ and so $\mathcal{C}$ and $\mathcal{S}$. This axis, (g_c, g_s), is defined by the vector

$$u = W^{-1}(g_c - g_s), \tag{11}$$

where $M = W^{-1}$ can be regarded as a metric. Actually, a new image, I representated by the point p will be said to belong to $\mathcal{C}$, 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 belongs $\mathcal{C}$ if and only if

$$p.u = pM(g_c - g_s) > T, \tag{12}$$

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

3.2 Learning step

For training our classifier, we use 2000 images from a database of about 100,000 JPEG images downloaded from the web, notably *https://www.worldprints.com* in 2000. No size, JPEG quality factor and color or grayscale discriminations are made. The sample is as close as possible as the *natural* population of JPEG images we can found on the internet.

First, we use the Multiple Embedding Method to compute $\mathcal{V}_c$ and $\mathcal{V}_s$, as described previously, for different lengths l of random messages to embed. We took $l = 10, 50, 100, 200, 400, 600, 800$ bytes. As shown in figure 5, we can represent the statistics vector in a 3D space.

For each steganographic algorithm and l, we determine the discriminant factor u for the metrics W^{-1} and V^{-1} as defined in section 3.1. Then, we obtain the detection curves as a fonction of the threshold, as shown in figure 6. Finally, we determine the optimal parameters, reported in table 2, for Outguess, F5 and JPHide.

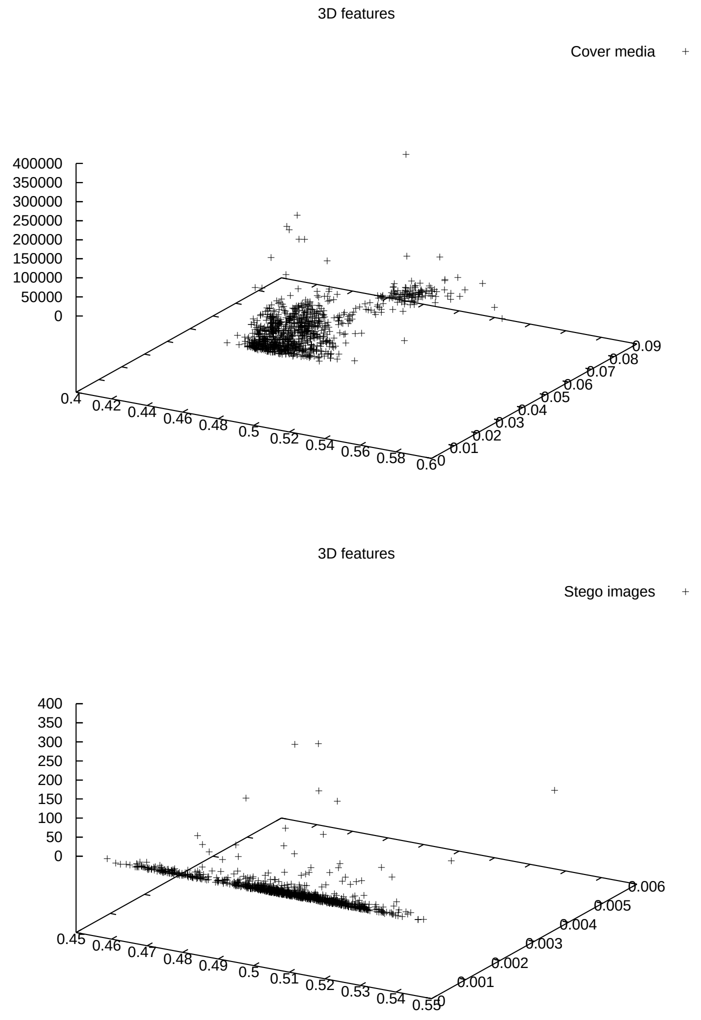

Fig. 5. Statistic vectors for JPHide and $l = 10$ bytes. On the top $\mathcal{V}_c$ and on the bottom $\mathcal{V}_s$.

3.3 Wild detection step

For testing the performances of our technique, we tested it with 2,000 randomly choosen images, including 1,000 stego images and 1,000 cover media, for an embedding rate ρ from 10^{-6} to 10^{-1}. These results are summarized in the figure 7. Two main conclusions can be drawn when observing these results. First, the the MEM seems to be very efficient, particulary when the embedding rate is low. That means that, we are able to detect efficiently stego images with only

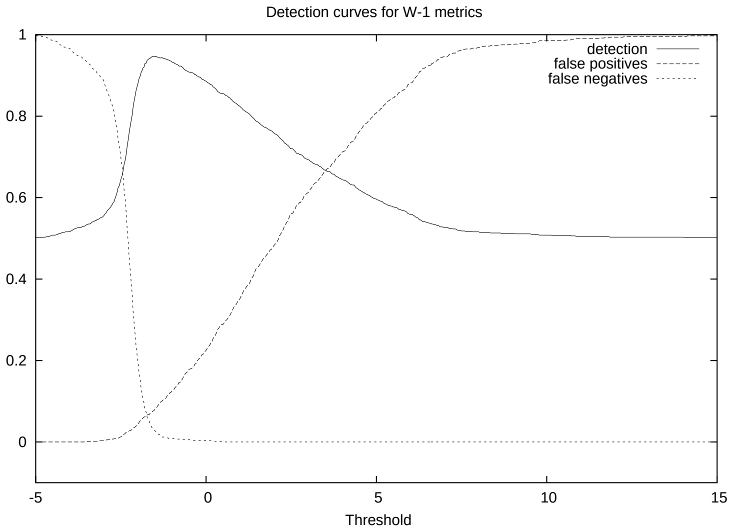

Fig. 6. Detection curves for Outguess and $l = 50$ bytes for metrics W^{-1}.

	Outguess	F5	JPHide
Metric	W^{-1}	W^{-1}	W^{-1}
threshold	-1.527165	-0.674345	-0.839874
l	50 bytes	10 bytes	10 bytes
u	$\begin{pmatrix} -3.897063E+01 \\ +2.299124E+02 \\ +2.046336E-04 \end{pmatrix}$	$\begin{pmatrix} -1.779760E+01 \\ +1.365236E+02 \\ +3.349051E-05 \end{pmatrix}$	$\begin{pmatrix} -1.143656E+01 \\ +1.472741E+02 \\ -7.731891E-06 \end{pmatrix}$

Table 2. Optimal parameters for Outguess, F5 and JPHide.

1 byte embedded. Secondly, the detection rate appears to be be constant and independent of ρ. More precisely, we observed what follows.

- The rate detection for Outguess is 93%, the false positive error rate 10% and the false negative error rate 3.8%.
- The rate detection for F5 is 88.4% , the false positive error rate 16.6% and the false negative error rate 6.6%.
- The rate detection for JPHide is 97.7%, the false positive error rate 3.7% and the false negative error rate 0.8%.

Obviously, these results depend on the distribution of cover media and stego images, but they give us a lower bound of the dectection rate. All the worste cases are obtained with sets only composed of cover media. So, for Outguess detection rate is higher than 90%, for F5 higher than 83.4% and for JPHide higher than 96.3%, whatever the distribution of cover media and stego images is.

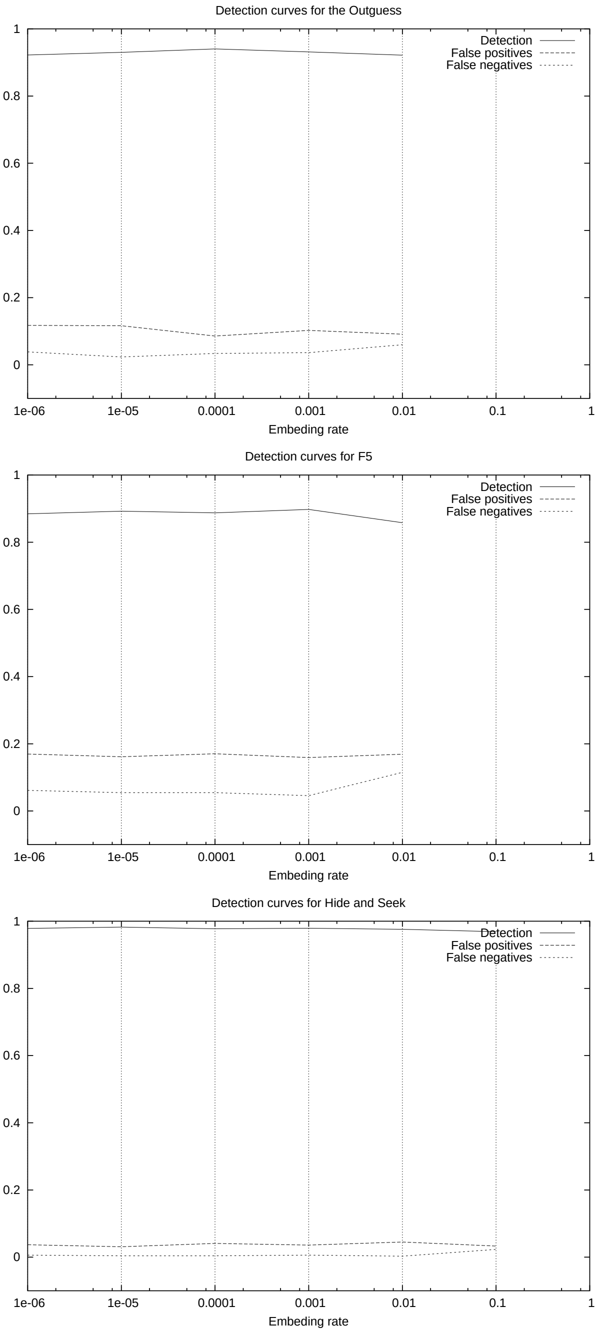

Fig. 7. The detection curves for Outguess, F5 and JPHide.

Conclusion

We have proposed a new approach for JPEG steganalysis which is based on statistics of the compressed frequency domain and pointed out new features to detect steganographic contents. This approach can be justified according to the assertion that it is hard for steganographic algorithms to preserve at the same time statistics in the spatial domain, in the frequency domain and in the compressed frequency domain. Moreover, we benefit from statistical deviation of the entropy of the binary output stream. We combined these new statistic features with the Multiple Embedding Method to design an efficient Fisher discriminant based classifier. 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 steganalyser with very high and constant detection rates, as illustrated in section 3.3. The experimental results show that our steganalysis scheme is able to efficiently detect the use of Outguess, F5 and JPHide and JPSeek, even if the embedding rate is very low ($\approx 10^{-6}$).
In future research, we will try to improve the efficiency of our classifier first using Support Vector Machines, then finding new features in the compressed frequency domain and finally combining it with detectors working in other domains. We are going working on generalizing our approach to universal steganalysis, too.

Acknowledgments

We want to warmly thank Emmanuel Mayer for many useful advices on implementation and Franck Landelle for his accurate knowledge of statistics. Many discussions with them have greatfully improved the quality of this work. We also thank Reynald Lercier for his comments on the paper.

References

1. I. Avicibaş, N. Memon, B. Sankur: *Steganalysis based on Image Quality Metrics.* SPIE Vol. 4314, San Jose, California, USA, 2001.
2. C. W. Brown, B. J. Shepherd: *Graphics File Formats, reference and guide.* Manning, 1995.
3. R. Chandramouli, M. Kharrazi, N. Memon: *Image Steganography and Steganalysis: Concepts and Pratice.* International Workshop on Digital Watermarking, Seoul, October 2003.
4. H. Farid: *Detecting Hidden Messages Using Higher-Order Statistical Models.* International Conference on Image Processing (ICIP), Rochester, NY, 2002.
5. J. Fridrich, M. Goljan, D. Hogea: *Steganalysis of JPEG Images: Breaking the F5 Algorithm.* 5th Information Hiding Workshop, pp. 310-323, Noordwijkerhout, The Netherlands, 7-9 October 2002.
6. J. Fridrich, M. Goljan, D. Hogea: *New Methodology for Breaking Steganographic Techniques for JPEGs.* Proc. SPIE Electronic Imaging, pp. 143-155, Santa Clara, CA, January 2003.

7. J. Fidrich: *Feature-Based Steganalysis for JPEG Images and its Implications for Future Design of Steganographic Schemes.* 6th Information Hiding Workshop, LNCS, vol. 3200, Springer-Verlag, pp. 67-81, 2004.
8. J. Fridrich, T. Pevny: *Multiclass Blind Steganalysis for JPEG Images.* Proc. SPIE Electronic Imaging, Photonics West, January 2006.
9. J. J. Harmsen, W. A. Pearlman: *Kernel Fisher Discriminant for Steganalysis of JPEG Hiding Methods.* ACM Multimedia and Security, New York August 1-2, 2005.
10. A. Latham: *Steganography: JPHIDE and JPSEEK, 1999.* http://linux01.gwdg.de/∼alatham/stego.html
11. G. Lin, C. H. Yeh, C. C. Jay Kuo: *Data hiding domain classification for blind image steganalysis.* Proceedings of the 2004 IEEE International Conference on Multimedia and Expo, ICME 2004, pp. 907-910, Taipei, Taiwan, 27-30 June 2004.
12. S. Lyu, H. Farid: *Detecting Hidden Messages Using Higher-Order Statistics and Support Vector Machines.* 5th International Workshop on Information Hiding, Noordwijkerhout, The Netherlands, 2002.
13. S. Lyu and H. Farid: *Steganalysis Using Color Wavelet Statistics and One-Class Support Vector Machines.* SPIE Symposium on Electronic Imaging, San Jose, CA, 2004.
14. S. Lyu, H. Farid: *Steganalysis Using Higher-Order Image Statistics.* IEEE Transactions on Information Forensics and Security, Number 1, pp. 111-119, 2006.
15. H. Feistel: *Cryptography and computer privacy.* Scientific American, vol. 228, Number 5, pp. 15-23, 1973.
16. N. Provos: *Defending Against Statistical Steganalysis.* 10th USENIX Security Symposium, Washington, DC, 2001.
17. G. Saporta: *Probabilité, analyse des données et statistiques.(in french).* Technip, 1990.
18. G. Simmons: *Prisoner' problem and the subliminal channel.* Crypto'83- Advances in Cryptology, pp. 51-67, 1984.
19. G. W. Wallace: *The JPEG Still Picture Compression Standard.* Communications of the ACM, Number 34, pp. 30-44, April 1991.
20. A. Westfeld: *F5 - A Steganographic Algorithm. High Capacity Despite Better Steganalysis* Proceedings of Information Hiding. 4th International Workshop, IH'01, LNCS 2137, pp. 289-302, Pittsburgh, USA, April 2001.