A Comparative Study of ± 1 Steganalyzers

Giacomo Cancelli [#1], Gwenaël Doërr [*2], Mauro Barni [#3], and Ingemar J. Cox [*4]

[#] *University of Siena – Information Engineering Department*
via Roma, 56 - 53100 Siena, Italy
[1] giacomo.cancelli@unisi.it
[3] barni@dii.unisi.it

[*] *University College London – Adastral Park Postgraduate Campus*
Ross Building 2, Martlesham IP5 3RE, UK
[2] g.doerr@adastral.ucl.ac.uk
[4] ingemar@ieee.org

***Abstract*—We compare the performance of three steganalysis system for detection of ± 1 steganography. We examine the relative performance of each system on three commonly used image databases. Experimental results clearly demonstrate that both absolute and relative performance of all three algorithms vary considerably across databases. This sensitivity suggests that considerably more work is needed to develop databases that are more representative of diverse imagery.**

In addition, we investigate how performance varies based on a variety of training and testing assumptions, specifically (i) that training and testing are performed for a fixed and known embedding rate, (ii) training is performed at one embedding rate, but testing is over a range of embedding rates, (iii) training and testing are performed over a range of embedding rates. As expected, experimental results show that performance under (ii) and (iii) is inferior to (i). The experimental results also suggest that test results for different embedding rates should not be consolidated into a single score, but rather reported separately. Otherwise, good performance at high embedding rates may mask poor performance at low embedding rates.

I. Introduction

Steganography is commonly framed as the prisoners' problem [1]. In this context, two prisoners, Alice and Bob, wish to exchange secret messages. The prison Warden, Eve, permits the prisoners to engage in benign communications, but all communications are first inspected by the Warden to ensure that the messages are indeed benign, i.e. contain no hidden message. If the Warden suspects that the communication contains a covert message, then the Warden will prevent the communication between the prisoners. If the Warden's tests indicate that the communication does not contain a covert message, then the message is forwarded, unchanged, to the recipient[1]. Benign content, e.g. images, are referred to as cover texts (cover images). Content that contains a covert message is referred to as stego text (stego image). Alice's and Bob's goal is to develop steganographic algorithms that are undetectable, which is very different from the watermarking requirement of imperceptibility. And the Warden's goal is to develop increasingly sophisticated tests to detect covert communication and thereby thwart Alice and Bob.

[1]In this case the Warden is said to be passive. The alternative case is that of an active Warden, in which Alice's message may be modified, e.g. lossy compression, prior to forwarding to Bob.

Steganography has received considerable interest during the last few years, especially after anecdotal reports alleged that this technology was used by terrorist and organized crime organizations. The concern is that multimedia content commonly transmitted over the Internet could be used as a cover to convey hidden, steganographic information. To respond to this concern, research effort has been directed towards the design and development of efficient steganalysis tools. The objective of steganalysis is to detect the use of steganography. Steganalysis tools are commonly categorized as either targeted or blind. Targeted steganalysis seeks to detect the use of a known steganographic algorithm, e.g. ± 1 embedding. Blind steganalysis seeks to detect a range of steganographic algorithms, possibly including previously unknown algorithms.

A wide variety of steganalysis algorithms have recently been proposed. However, their performances is often reported under slightly different conditions, which hampers an accurate comparison of algorithms. To illustrate this problem, we focus on the evaluation of steganalysis techniques for the detection of ± 1 steganography applied to still images. Section II first reviews three state-of-the-art steganalyzers used to detect ± 1 steganography. Section III then provides a detailed description of the procedure applied to evaluate these steganalyzers and emphasizes the different parameters which may have an impact on classification performance e.g. the test and training databases and the embedding rate. Section IV details experimental results that clearly highlight very high variability in the classification performance depending on the experimental parameters. Finally, Section V provides a summary and discussion of results.

II. LSB Matching Steganalysis

Least Significant Bit (LSB) matching steganography, also referred to as ± 1 embedding, is a slightly more sophisticated version of LSB embedding. It can be mathematically described as follows:

$$p_s = \begin{cases} p_c + 1, & \text{if } b \neq \mathrm{LSB}(p_c) \text{ and } \big(\kappa > 0 \text{ or } p_c = 0\big) \\ p_c - 1, & \text{if } b \neq \mathrm{LSB}(p_c) \text{ and } \big(\kappa < 0 \text{ or } p_c = 255\big) \\ p_c, & \text{if } b = \mathrm{LSB}(p_c) \end{cases} \quad (1)$$

where p_s (resp. p_c) denotes a pixel value in the range $0 \dots 255$ in the stego image (resp. cover image), b is the message bit to be hidden, and κ is an i.i.d. random variable with uniform distribution on $\{-1, +1\}$[2]. Depending on the length of the message to be hidden, the whole LSB plane could be used as a cover or only a portion of it. The ratio between the size of the LSB plane and the length of the message is referred to as the *embedding rate*, denoted by ρ.

Although LSB replacement steganography is known to be relatively easy to detect, LSB matching has proved to be a much more difficult problem. A series of steganalyzers have been developed for this purpose and the objective of this paper is to provide a fair comparison between those different approaches. To this end, three state-of-the-art steganalyzers are briefly reviewed in the next subsections. They can be roughly considered as sharing a common architecture, namely (i) feature extraction in some domain and (ii) Fisher Linear Discriminant (FLD) analysis to obtain a 2-class classifier.

A. High Order Statistics of the Stego Noise

Since LSB matching steganography adds or subtracts 1 to a subset of pixel values, it can be modeled as the addition of high frequency noise. In [2], Goljan *et al.* suggested estimating this stego noise by applying some denoising techniques to the detail bands of a first order wavelet decomposition of an image, and subsequently characterizing this estimated noise with a collection of central absolute moments defined as follows:

$$m_{\mathbf{b}}^{p} = \frac{1}{|\mathcal{I}|} \sum_{(i,j)\in\mathcal{B}} |\mathbf{n}_{\mathbf{b}}(i,j) - \bar{\mathbf{n}}_{\mathbf{b}}|^{p} \qquad (2)$$

where $\mathbf{n}_{\mathbf{b}}$ is the estimated stego noise in subband $\mathbf{b}$, $\bar{\mathbf{n}}_{\mathbf{b}}$ is its mean value, and $\mathcal{B}$ some bidimensional index covering all the samples in the subband. Due to its construction, this system is referred to as Wavelet Absolute Moment (WAM) steganalysis. The authors suggested to use a feature vector $\mathbf{f}_{\mathrm{WAM}}$ consisting of 27 moments (9 per band) and noted that this method is not specific to ± 1 steganography, i.e. it is a blind steganalysis technique.

B. Center of Mass of the Histogram Characteristic Function

In [3], Harmsen and Pearlman noted that LSB matching using an embedding rate ρ induces a low-pass filtering of the intensity/color histogram of the image with the following kernel:

$\rho/4$	$1-\rho/2$	$\rho/4$

This means that the histogram of a stego Work contains less high-frequency power than the histogram of the corresponding cover image, thus shifting the center of mass of the histogram's spectrum toward the origin. Using this feature to discriminate stego from non-stego content, the authors observed better classification performances for RGB images than with grayscale images.

[2]Note that this strategy may affect bit-planes other than the LSB plane. For example, if the secret bit is a "0", and the original 8-bit pixel value is 01111111, then incrementing this value results in 10000000.

Ker suggested that this difference in performances could be due to a lack of sparsity in the histogram of grayscale images [4]. He then proposed to use a two-dimensional adjacency histogram which tabulates how often each pixel intensity is observed next to another. He showed that LSB matching steganography also reduces to low-pass filtering the adjacency histogram and defined a center of mass of the adjacency histogram characteristic function. In addition, to reduce the variability of this feature across images, Ker recommended computing the same center of mass using a downsampled version of the image. The final scalar feature $\mathbf{f}_{\mathrm{2D-HCFC}}$ to be used during classification is then obtained by computing the ratio between these two values. This steganalyzer, referred to as 2D-HCFC, is targeted for ± 1 steganography.

C. Amplitude of Local Extrema

Based on the same observation that LSB matching steganography is equivalent to low-pass filtering the intensity histogram, Zhang *et al.* [5] chose to focus their attention on the local extrema of the histogram. The filtering operation will indeed reduce the amplitude of local extrema. As a result, they rely on the sum of the amplitudes of all local extrema in the histogram to distinguish stego content from non-stego content.

Inspired by Ker's work, Cancelli *et al.* [6] subsequently extended this strategy to four 2D adjacency histograms (adjacency in the horizontal, vertical, main diagonal and minor diagonal direction). Combined with the previous features, this results in a 10-dimensional feature vector, $\mathbf{f}_{\mathrm{ALE}}$. Due to its construction, this steganalysis system is referred to as Amplitude of Local Extrema (ALE). ALE is also targeted toward LSB matching steganography.

III. Experimental Setup

This section provides a detailed description of the experimental setup used in this study. It first describes the different databases used to obtain performances statistics and then reviews the experimental protocol for evaluating all steganalysis systems.

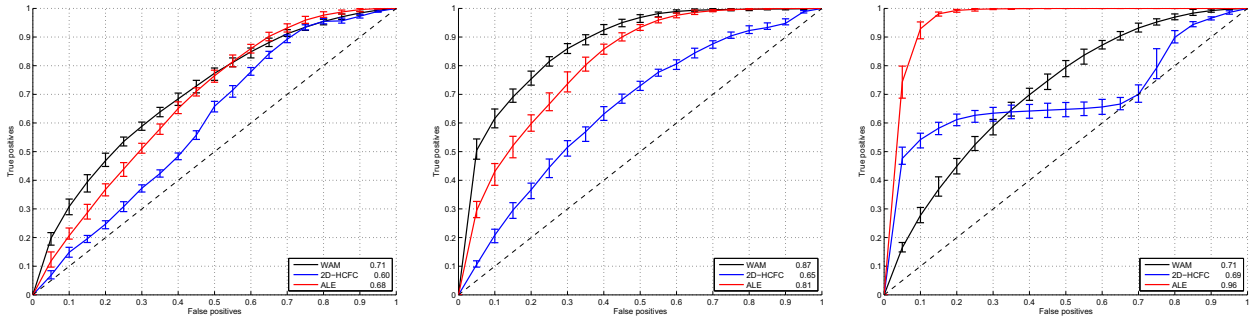
A. Databases

This study used three different databases that have been previously used in the context of steganography and watermarking. The three databases not only contain different images, but, more importantly, the image sources are significantly different, as discussed shortly. The motivation for using more than one database was to determine any variability in performance across databases. A fourth database was created as the concatenation of the three primary databases[3].

The four image databases are:

1) **NRCS Photo Gallery:** This image database is provided by the United States Department of Agriculture [8]. It contains 2,375 photos related to natural resources and conservation from across the USA, e.g. landscape,

[3]To encourage the use of these databases, the authors have made them accessible on their website [7].



(a) NRCS database (b) Camera images database (c) Corel database

Fig. 1. Training and testing on individual databases: each figure depicts the vertically averaged ROC curves for the three steganalyzers under study (WAM, 2D-HCFC, ALE). LSB matching has been applied with an embedding rate, $\rho = 0.5$ bits per pixel. The error bar at each sampling point indicates the minimum and maximum true positive value observed at a given false positive value. The number in the legend indicates the area under the ROC curve (AUC).

cornfields, etc. Typically, the image formats are in 32-bit CMYK space color and in high resolution, i.e. 1500×2100. Unfortunately, there is no indication of how these photos were acquired. This image database has first been used in [4].

2) **Camera Images:** This image database is a collection of 3,164 images captured using 24 different digital camera (Canon, Kodak, Nikon, Olympus and Sony) by researchers from Binghamton University, NY, USA. It includes photographs of natural landscapes, buildings and object details. All images have been stored in a raw format i.e. the images have *not* undergone lossy compression. A subset of these images was previously used in [2].
3) **Corel database:** The Corel image database consists of a large collection of uncompressed images [9]. They include natural landscape, people, animals, instruments, buildings, artwork, etc. Although there is no indication of how these images have been acquired, they are very likely to have been scanned from a variety of photos and slides. A subset of 6,185 images has been extracted from the database with dimension 512×768.
4) **Combined database:** A fourth database was created by concatenating 2375 randomly selected images from each of the three databases.

Where necessary, all images have been converted to 8 bit-depth grayscale. Moreover, a central cropping operation of size 512×512 was applied to all images to obtain images of the same dimension across all three databases. Cropping was preferred over resizing, in order to avoid introducing artifacts due to resampling with interpolation.

B. Experimental Procedure

For each one of the four available image databases (NRCS, Camera, Corel, Combined), the following procedure was performed for each one of the three steganalyzers under study (WAM, 2D-HCFC, ALE):

1) Apply LSB embedding steganography with embedding rate ρ to all images in the database $\mathcal{D}$ to obtain the database of stego images $\mathcal{D}^*$;
2) Separate both databases $\mathcal{D}$ and $\mathcal{D}^*$ into a training set, $\{\mathcal{D}(\mathcal{I}), \mathcal{D}^*(\mathcal{I})\}$, and a test set, $\{\mathcal{D}(\bar{\mathcal{I}}), \mathcal{D}^*(\bar{\mathcal{I}})\}$, where $\mathcal{I}$ is a subset of the image indexes and $\bar{\mathcal{I}}$ is its complement. The size of the training set was set to be equal to 20% of the database size;
3) For the steganalyzer under test, compute the associated feature vector for all images in the training set and perform FLD analysis [10] to obtained the trained projection vector $\mathbf{p}$;
4) For the steganalyzer under test, compute the associated feature vector for all images in the test set, and project the feature vector onto $\mathbf{p}$;
5) Compare the resulting scalar values to a threshold τ and record the probabilities of false positives and true positives for different values of the threshold in order to obtain the Receiver Operating Characteristic (ROC) curve of the system.

Steps 2 through 5 are repeated 20 times for cross-validation [10] and the ROC curves are vertically averaged [11]. That is, for a fixed false positive value, the corresponding true positive rates for each of the 20 curves are averaged. The confidence level at each false positive point depicted in the resulting curves indicates the minimum and maximum true positive rates from the set of ROC curves.

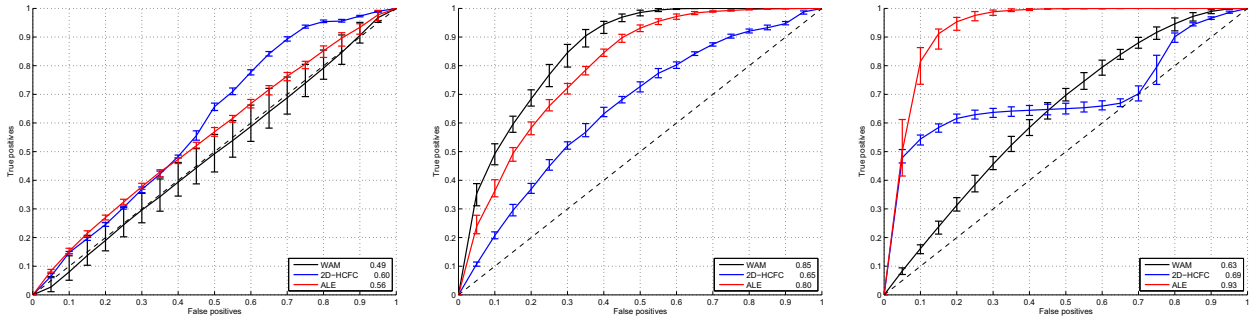
The overall performance of the steganalyzer can be summarized by computing the area under the ROC curve (AUC) [11]. An AUC value close to 1 indicates excellent discrimination, while a value close to 0.5 indicates poor discrimination.

IV. Experimental Results

In Section IV-A, we report experimental results that examine the variability in performance across databases. Section IV-B then discusses how performance is affected when the embedding rate is not assumed to be known.

A. Impact of Image Databases

For comparison purposes, we first examine the performance of the three steganalyzers under study on each of the individual databases. In this case, training is performed on the individual



(a) Testing on NRCS database (b) Testing on Camera images database (c) Testing on Corel database

Fig. 2. Training on Combined dataset of 2375 NRCS + 2375 Camera + 2375 Corel images. Training was performed with 20% of the joint set. Testing was performed with the remaining images in each databases.

databases, and the embedding rate ρ set to 0.5. Experimental results are depicted in Figure 1. Similar behaviour was observed for other embedding rates but data is omitted for brevity and clarity. It is clear from Figures 1 that the absolute performance of the three algorithms varies considerable across the three primary image databases. In addition, the relative performance is also seen to vary.

For the NRCS and Camera testsets, the WAM steganalyzer exhibits best performance. However, even across these two testsets, the absolute performance varies significantly. For example, for a false positive rate of 10%, the WAM algorithm has a true positive rate of 30% and 60% for NRCS and Camera respectively. There are similar variations for the other two algorithms. For the Corel database, the ALE steganalyzer performs much better. Interestingly, for the Corel database, we observe very strange behaviour for the 2D-HCFC algorithm, where the true positive rate remains almost constant as the false positive rate increases from 0.2 to 0.7.

The experimental results clearly highlight that the steganalyzers perform noticeably differently on different databases. Thus, the three databases cannot be exchanged interchangeably. This variability not only makes comparison of steganalysis algorithms tested on different databases impossible, but may have significant affect on performance if the database used for training does not adequately represent the class of images observed during testing. Certainly, a steganalyzer trained on one database and tested on another may exhibit degraded performance since the training database is unlikely to be representative of the test database. When deployed in real life, steganalyzed images will not be taken from some database that the steganalyzer can use for training. Consequently, systems need to be trained on large database that accurately model the class of images under test.

To approximate the deployment scenario, we repeated the same experiment using the combined database for training. Classification results are reported in Figures 2 for each separate database *that constitute the combined database*, rather than as a single curve. In this way, we are still able to observe performance differences across the various classes of imagery. As expected, the performance of the steganalyzers generally degrades since training is no longer database specific. However, this degradation is more or less significant depending on the steganalyser. Indeed, the AUC values for the 2D-HCFC algorithm remain almost unchanged, whereas the performance of WAM is noticeably degraded, especially with the Corel and NRCS databases. In the latter case, the relative performance between steganalyzers is strongly affected, with WAM becoming the worst algorithm whereas it was the best in Figure 1(a).

B. *Impact of Unknown Embedding Rates*

The previous results were computed in the case where both training and testing were conducted for a known, fixed embedding rate ρ. However, in any real operating scenario, the steganalyst is unlikely to have knowledge of the embedding rate used by the steganographer. Thus, it is necessary to design a steganalysis algorithm that performs well for a variety of embedding rates. Figure 3 shows the performance of the three steganalyzers when trained using embedding rates of $\rho = 0.2$, 0.5 and 1.0 bit per pixel (bpp). For each of these three trained classifiers, test results are reported using stego content with embedding rates of $\rho = 0.2$, 0.5 and 1.0 bpp[4]. Each row represents one of the three training conditions and each column represents performance on one of the three different test sets.

One would expect that a steganalyzer trained on data with an embedding rate of $\rho = 0.2$ would exhibit improved performance on test sets that used a higher embedding rate. However, while this is generally true, it is not always the case. For example, let us consider the performance of WAM in the first row of Figure 3. It is clear that performance when testing on $\rho = 0.2$, the same as for training, is actually better than when tested with a dataset with an embedding rate of $\rho = 1.0$. The same observation holds for WAM when trained at 0.5 bpp, where for the 0.5 bpp test set the AUC=0.68, but for the 1.0 bpp test set, the AUC=0.63. Clearly, performance is always best when training and testing conditions match, but the degradation in performances in case of mismatch varies from one system to the other. Whereas WAM is very sensitive to such mismatch, 2D-HCFC is completely immune to it since

[4]More exhaustive tests were conducted over a wider range of embedding rates. However space limitations preclude including all experimental results.

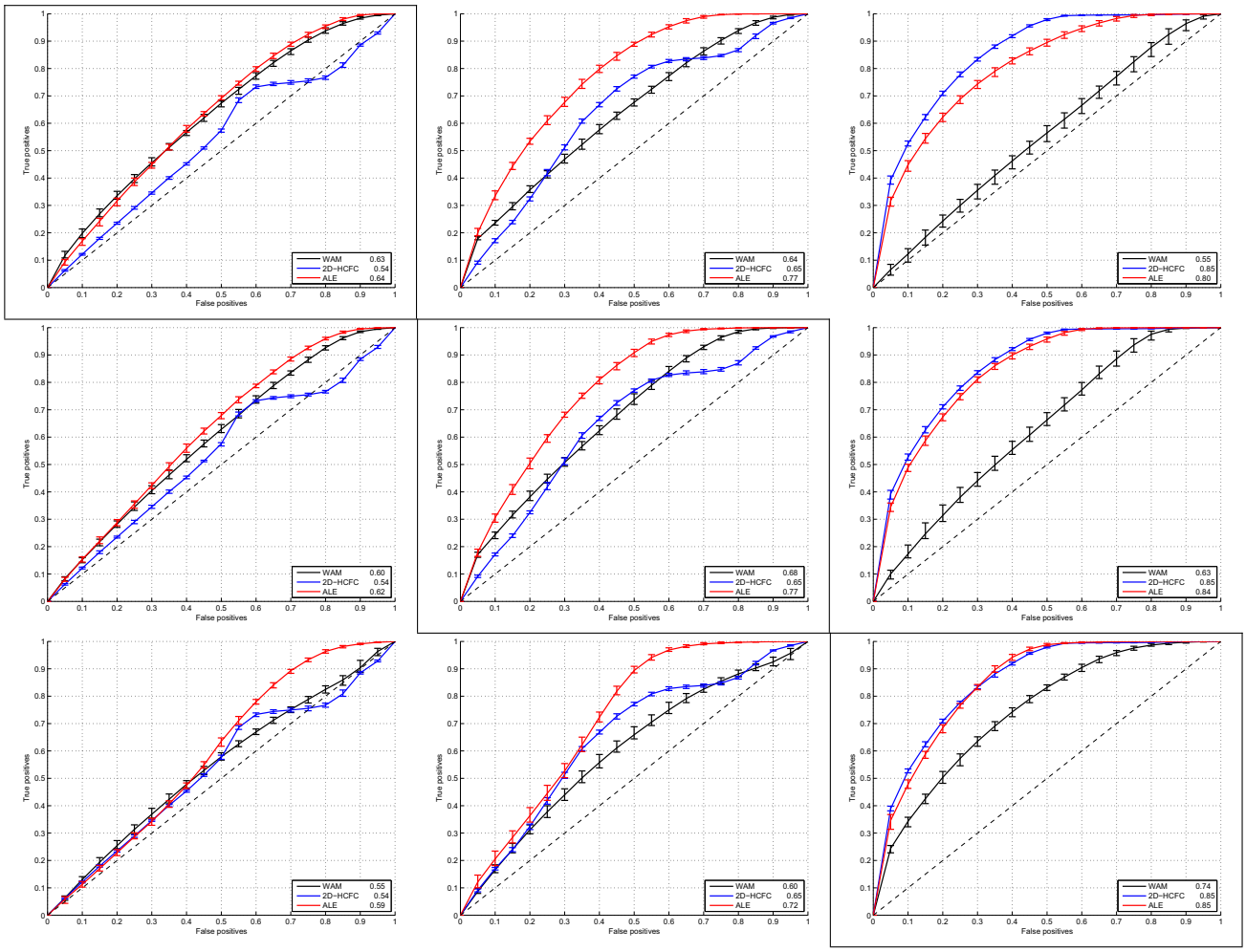


Fig. 3. Classification results on the combined database. Training at 0.2 bpp (top row), 0.5 bpp (middle row) and 1 bpp (bottom row) – testing with embedding at 0.2 bpp (left column), 0.5 bpp (middle column) and 1 bpp (right column). Rows indicate the training regime whereas columns indicate the embedding rate testing regime. Bounding boxes indicate setup with matching training and testing regimes.

it uses a scalar feature vector and has therefore no training phase *per se*.

In practice, things are even more complicated, as it is also unlikely that steganographers will embed at a fixed embedding rate. Thus, it is also instructive to consider the performance of the steganalyzers when trained with a fixed embedding rate, but tested with a range of embedding rates as illustrated by Figure 4. Training is performed using the combined database for a fixed embedding rate of either 0.2, 0.5 and 1 bpp. Testing is conducted with using a test set that includes all three embedding rates, 50% of images being cover images and the remaining 50% stego images with embedding rates uniformly distributed across 0.2, 0.5 and 1 bpp. It appears that the performance, as measured by the AUC value, increases as the training embedding rate increases. In other words, it seems that, perhaps contrary to expectation, it is better to train with a high embedding rate (e.g. 1 bpp) rather than a low embedding rate (e.g. 0.2 bpp). The explanation is that, since we are testing across a range of embedding rates that are uniformly distributed but training on only one embedding rate, performance is maximized if we train to detect stego Works that are most easy to detect, i.e. with a high embedding rate (1 bpp). This suggests that providing overall scores averaged over all test data may be misleading. It may be more informative to provide separate results for each embedding rate included in the test set.

Since a steganalysis system will be used to test images with different embedding rate, the steganalyst may also decide to train with a combination of embedding rates. Figure 5 illustrates this arrangement. Training has been conducted using the combined database with a training set that contains stego Works with embedding rates of 0.2, 0.5 and 1 bpp. On the other hand, testing is performed with separate tests sets for each of the three embedding rates. As expected, performance usually improves as the embedding rate (for testing) increases. Only WAM seems to deviate from this rule.

V. Summary and Discussion

Our experimental results demonstrate that the performance of current state-of-the-art steganalyzers for detection of ± 1 steganography is highly sensitive to the databases used for

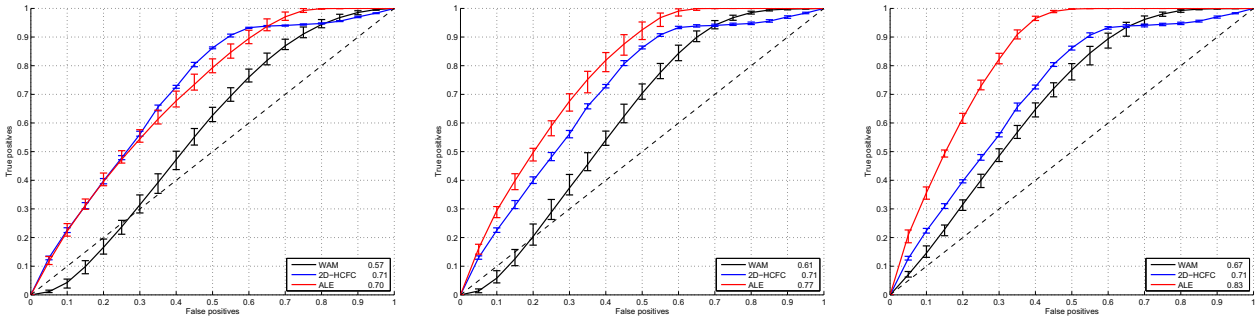


Fig. 4. Classification results on the combined database. Training at 0.2 bpp (left), 0.5 bpp (middle) and 1 bpp (right) – testing with all embedding rates.

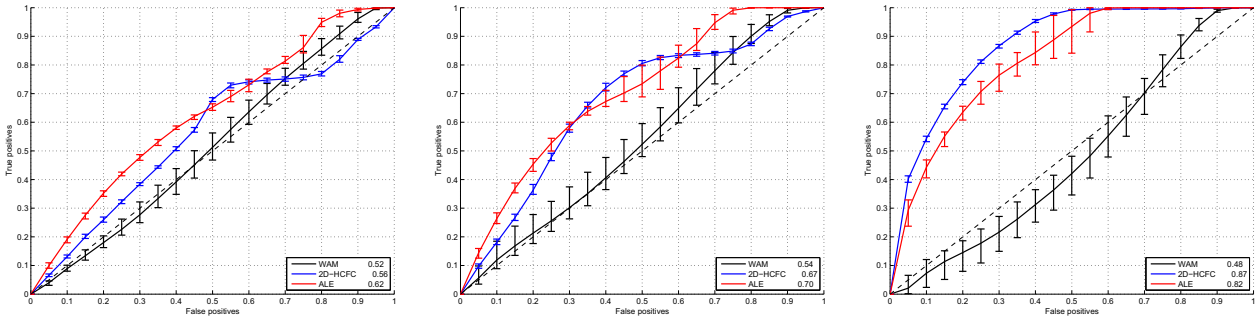


Fig. 5. Classification results on the combined database – Training with all embedding rates, testing at 0.2 bpp (left), 0.5 bpp (middle) and 1 bpp (right).

training and testing. This is not necessarily a problem of the algorithm, but rather, highlights the need for better databases and a detailed characterization of the classes of imagery that they represent. It is only necessary for a steganalysis algorithm to work within the class of imagery that the steganalyst observes. However, at present this is not discussed and there has been no work at describing and characterizing possible operating scenarios and the associated classes of imagery. Meanwhile, we believe it will benefit the research community to agree on one or more standardized testsets to be used for comparative purposes.

Our experimental results also suggest that it is not sufficient to report performance for a single embedding rate, used for both training and testing. This does not represent a real operating scenario where the embedding rate for testing is very likely to be unknown. Experimental results indicate that it cannot be assumed that if an algorithm performs well when trained and tested with a low embedding rate, then the performance will be better when tested (not trained) for higher embedding rates. This is certainly not the case for the WAM algorithm. When training and testing across a variety of embedding rates, our experimental results suggest that summarizing detection rates can be misleading - good performance at high embedding rates may mask poor performance at low embedding rates. We therefore recommend that test results be reported for each tested embedding rate.

Our comparison revealed that no single steganalysis algorithm was consistently superior. This suggests that improved performance may be obtained by combining the results of multiple algorithms.

ACKNOWLEDGMENTS

The authors would like to thank J. Fridrich and M. Goljan for providing the source code of the WAM steganalyzer and the image databases they are using. Giacomo Cancelli and Mauro Barni were partially supported by the Italian Ministry of Research and Education under FIRB project no. RBIN04AC9W.

REFERENCES

[1] G. J. Simmons, "The prisoners' problem and the subliminal channel," in *Advances in Cryptology: Proceedings of CRYPTO'83*. Plenum Pub Corp, 1984, pp. 51–67.

[2] M. Goljan, J. Fridrich, and T. Holotyak, "New blind steganalysis and its implications," in *Security, Steganography, and Watermarking of Multimedia Contents VIII*, ser. Proceedings of SPIE, vol. 6072, January 2006, pp. 607 201–1.

[3] J. Harmsen and W. Pearlman, "Steganalysis of additive noise modelable information hiding," in *Security and Watermarking of Multimedia Contents V*, ser. Proceedings of SPIE, vol. 5020, January 2003, pp. 131–142.

[4] A. D. Ker, "Steganalysis of LSB matching in grayscale images," *IEEE Signal Processing Letters*, vol. 12, no. 6, pp. 441–444, June 2005.

[5] J. Zhang, I. J. Cox, and G. Doërr, "Steganalysis for LSB matching in images with high-frequency noise," in *Proceedings of the IEEE Workshop on Multimedia Signal Processing*, October 2007, pp. 385–388.

[6] G. Cancelli, I. J. Cox, and G. Doërr, "Improved LSB matching steganalysis based on the amplitude of local extrema," in *IEEE International Conference on Image Processing*, October 2008.

[7] G. Doërr. (2007) Image database for steganalysis studies. [Online]. Available: http://www.adastral.ucl.ac.uk/ gwendoer/steganalysis

[8] United States Department of Agriculture. (2002) Natural resources conservation service photo gallery. [Online]. Available: http://photogallery.nrcs.usda.gov

[9] Corel Corporation, "Corel stock photo library 3," Ontario, Canada.

[10] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. Wiley-Interscience, 2001.

[11] T. Fawcett, "ROC graphs: Notes and practical considerations for researchers," HP Laboratories, Tech. Rep., March 2004.