Advantages of Using Feature Selection Techniques on Steganalysis Schemes

Yoan Miche[1,2], Patrick Bas[1,2], Amaury Lendasse[1],
Christian Jutten[2], and Olli Simula[1]

[1] Helsinki University of Technology - Laboratory of Computer and Information Science
P.O. Box 5400, FI-02015 HUT, Finland
[2] INPG - Laboratoire des Images et des Signaux,
INPG, 46 avenue Félix Viallet, 38031 Grenoble cedex, France

Abstract. Steganalysis consists in classifying documents as steganographied or genuine. This paper presents a methodology for steganalysis based on a set of 193 features with two main goals: determine a sufficient number of images for effective training of a classifier in the obtained high-dimensional space, and use feature selection to select most relevant features for the desired classification. Dimensionality reduction is performed using a forward selection and reduces the original 193 features set by a factor of 13, with overall same performance.

1 Introduction

Steganography has been known and used for a very long time, as a way to exchange information in an unnoticeable manner between parties, by embedding it in another, apparently innocuous, document. For example, during the 80's, Margaret Thatcher decided to have each word processor of the government's administration members changed with an unique word spacing for each, giving a sort of "invisible signature" [7] to documents. This was done to prevent the continuation of sensitive government information leaks.

Nowadays steganographic techniques are also used on digital contents. The online newspaper, Wired News, reported in one of its articles [4] on steganography that several steganographic contents have been found on websites with very large image database such as eBay.

Most of the time research about steganography is not as much to hide information, but more to detect that there is hidden information. This "reverse" part of the steganography is called steganalysis and is specifically aimed at making the difference between genuine documents, and steganographied – called stego – ones. Consequently, steganalysis can be seen as a classification problem where the goal is to build a classifier able to distinguish these two sorts of documents.

During the steganographic process, a message is embedded in an image so that it is as undetectable as possible. Basically, it uses several heuristics in order to guarantee that the statistics of the stego content are as close as possible to the statistics of the original one. Afterwards, steganalysis techniques classically

F. Sandoval et al. (Eds.): IWANN 2007, LNCS 4507, pp. 606–613, 2007.

use features extracted from the analysed image and an appropriately trained classifier to decide whether the image is genuine or not.

In this paper the Outguess algorithm proposed by Niels Provos in [9] is analysed. This algorithm, according to its author, is supposed to resist especially well to statistical attacks by work on Least Significant Bits of quantized DCT coefficients of JPEG compressed images. In practice, it is often used as a reference steganographic algorithm for performance comparison, eventhough it is highly detectable, as shown for example in [6,14].

In this work, the 193 image features proposed by Pevni and Fridrich in [12] have been used. Theses features consider statistics of JPEG compressed images such as histograms of DCT coefficients for different frequencies, histograms of DCT coefficients for different values, global histograms, blockiness measures and co-occurrence measures. They are an extension of an original set of 23 features [6].

Fridrich proposes afterwards to train a classifier according to the extracted features. Consequently a set of 193 features for each image of the database is obtained, giving an especially high dimensionality space for classifiers to work on. Earlier research about these high dimensionality spaces has shown that a lot of issues come out when the number of features is as high as the one we use.

2 Drawbacks of Performing Steganalysis in High-Dimensional Spaces with a Constrained Data-Set

The common term “curse of dimensionality” [2] refers to a wide range of problems related to a high number of features. Some details are given below about three inherent issues that occur in the framework of steganalysis, namely the need for data points (images), the increase of complexity and the lack of interpretability.

The need for data points: In the general case, in order for any tool to be able to analyze and find an underlying structure within the data, the number of needed points is growing exponentially with the dimension. Indeed, consider a d-dimensional unit side hypercube, the number of points needed to fill the Cartesian grid of step ϵ inside of it, is growing as $O((1/\epsilon)^d)$. Thus, using a common grid of step 1/10 and a dimension of 10, it requires 10^{10} points to fill the grid. In practice, steganalysis work makes often use of at least 10 to 20 dimensions, implying a “needed” number of images impossible to achieve. As a consequence, the feature space may be not correctly filled with data points, which can give wrong models when building classifiers, having to extrapolate for the missing images: thus, an estimation of the required minimum number of images has to be obtained, in order to have a reliable training of the selected model.

Increase of complexity: Computational time is another main reason. Nearest neighbours methods are usually implemented with a $O(d)$ dimension relationship, as for SVMs. Clearly, reducing the dimensionality by a significant order of magnitude gives much more achievable computational times. As a consequence, one can use more images and lower this “missing images” effect. Meanwhile, the finally chosen number of images should still remain within a reasonable range defined by these computational times. The best compromise between the number of images and the future reliability of our model has to be found.

Lack of interpretability: Eventhough the nearest neighbours classifiers keep good performance in high dimensions [3,8], other obvious problems of high dimensionality motivate the idea of feature selection. The interpretability is an important one: high performance can indeed be reached using the whole 193 features set for classification. Meanwhile, if looking for the weaknesses and reasons why these features react vividly to a specific algorithm, it seems rather impossible on this important set. Reducing the required number of features to a small amount through feature selection enables to understand better why a steganographic model is weak on these particular details, highlighted by the selected features. Indeed, steganalysis can tend to make the whole process aimed only at performance without possible interpretations, while having knowledge about the selected features gives interesting insights on the steganographic algorithm.

3 Methodology and Techniques Used

3.1 Classifiers and Appropriate Number of Images

The "appropriate" number of images (or at least the minimum required for the dimensionality of our data) should be determined. For this matter, a KNN classifier is used with a Monte-Carlo technique [11]. This enables to estimate the noise and give a confidence interval for our results. We randomly draw (without repetitions) a subset of the whole data set, and use the obtained classifier on it.

For our experiments, two different types of classifiers have mainly been used: the first one, KNN, for its overall good performance even in high dimensional spaces, but most of all, because it is computationally very fast. SVM was also chosen because it is among the classifiers giving the best results. Major drawback is of course the computational time. KNN classifier is a supervised distance-based classifier, proposed by Devijver and Kittler in [1], usually using the euclidean metric. It is based on a majority vote among the k nearest neighbours classes to assign the class of the new considered point. The SVM has been created by Vapnik [10] in 1963 and then improved more recently (1992,1995) by Boser, Guyon and Vapnik [5]. The original idea was to separate data using a hyperplane: this was a linear classifier. The extension of this method adds a non-linear part by the use of kernel functions.

3.2 Feature Selection Technique: Forward Selection

The forward selection algorithm is a greedy algorithm proposed in [13]; the algorithm selects one by one the dimensions, trying to find the one that combines best with the already selected ones. Even if its capacity to isolate efficient features is obvious, the forward technique has some drawbacks: in the case where two features would have a high dependency and be "unefficient" when alone but very good when put together, forward might not take these into account soon enough in the selection process. Nevertheless, the feature selection using forward has been showing very good results and seems to perform well on our feature set; this is presented in the next section.

3.3 Our Methodology

The three main points of the proposed methodology are detailed in the following. Fig. 1 illustrates the process. First is seeked a possibly good candidate for the number of images to use for training with the prepared database. Using a Monte-Carlo method on low numbers of images with both SVM and KNN, averaged plots are obtained. From it, a correct idea of a sufficient number of images for the later study can be obtained, as shown in the following experiments.

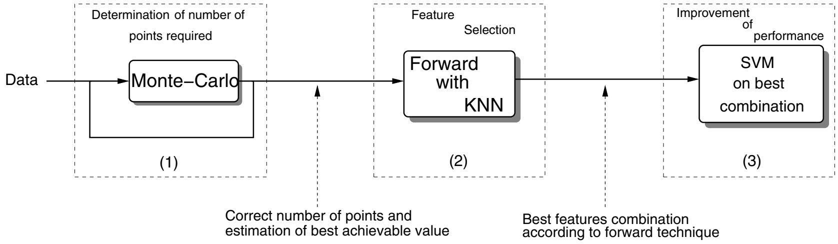
Determination of number of points required
Data
Monte–Carlo
(1)
Feature Selection
Forward with KNN
(2)
Improvement of performance
SVM on best combination
(3)
Correct number of points and estimation of best achievable value
Best features combination according to forward technique

Fig. 1. Schematic view of the proposed methodology: (1) An appropriate number of data points to work with is determined using a Monte-Carlo method for statistical stability; (2) The forward selection is performed using a KNN classifier; (3) A good feature set is selected and performance is improved using SVM.

Since KNN is the fastest classifier between the two presented, it is used for the next step with forward technique. This produces a ranking of the features showing how much each new feature contributes to the correct classification rate. The best features combination is selected. A SVM is finally used on this combination, to improve the performance and obtain the final best classification rate achieved.

4 Experiments, Results and Analysis

Our image base was constituted of 13 000 images of natural scenes, coming from 5 different digital cameras. Images are then all reduced to a size of 800 × 600 (multiples of 8) to avoid some possible block effects and artifacts due to JPEG recompression on another grid. At the same time, they are changed from their original colorspace to grayscale colorspace (256 gray levels).

A cropping operation to 512 × 512 follows, since our implementation of the extractor of Fridrich's 193 features works on 512 × 512 image blocks (powers of 2). In the end, the whole set of images is separated into two equal parts: one is kept genuine while the other one is steganographied with the Outguess algorithm at an embedding rate of 25%.

This choice of half steganographied and half genuine can be discussed as it does not reflects a real world situation. Meanwhile, the whole steganalysis process presented is designed to be used on one image at a time, determining whether it is genuine or not. Furthermore, this choice has been done to be able to compare performances with the steganalysis community current research.

For classification and test purposes, the training set has been made with at most 8000 images. Test set is composed of the remaining, that is 5000 images. The 193 features proposed by Fridrich are used as in [12].

4.1 Determination of Sufficient Number of Images

Presented first is the result of the evaluation of a sufficient number of images, as explained in the methodology. The Monte-Carlo is used on randomly taken subsets of 200 up to 2000 images with 10 iterations. Each model built – using KNN and SVM – is also evaluated on the test set of 5000 images.

A single point is evaluated with a randomly chosen set of 4000 images, since computational time becomes very important with such number of images. In practice, on Fig. 2 presenting the results of this study, the two cross-validation results (SVM and KNN) should not be strictly compared since they do not use the same number of images to validate the model: SVM uses a 10-fold cross-validation, while a Leave-One-Out (LOO) method is used for KNN. Test results are, on the other hand, comparable.

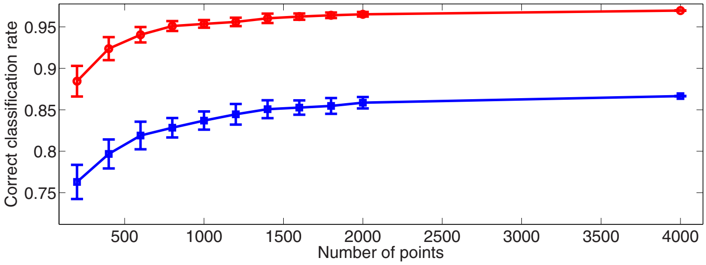

Fig. 2. Correct classification rate for SVM (circles, top curve) and KNN (squares, bottom curve) with associated variance

One can really see on these plots that an apparently sufficient number of images is over 2000, since the classification rate seems to increase exponentially slowly over this value. For the experiments, a bigger set of 4000 images has been chosen.

4.2 Forward Selection and Optimisation by SVM

Here, the results of the forward selection are presented shortly. As can be seen from Fig. 3, the whole process of forward selection is not fully achieved – for computational time reasons – since we do not go over 21 features. Meanwhile, as will be more detailled in the analysis part of these results, good performance is already performed before this value. Since the goal is to have the smallest possible feature set, while keeping average same performances, the forward selection could be stopped at this point.

Test and 10-fold cross-validation remain in a much thinner interval than for our only-KNN tryouts. Moreover, the performance gain with SVM is significative as expected and reaches up to 2% in the frame of these plots.

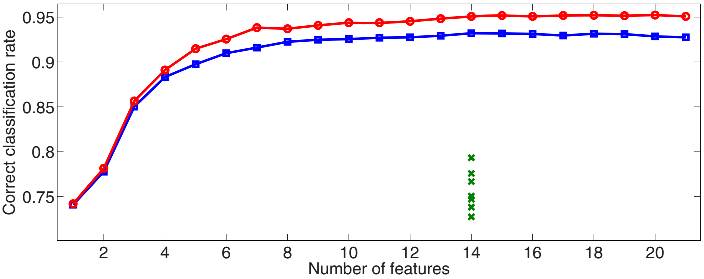

Fig. 3. Plot of the correct classification rate for SVM: 10-fold cross-validation (circles, top curve) and test (squares, bottom curve). Crosses are for performance for random sets of 14 features with a KNN classifier.

Table 1 presents the main values obtained using 193 features set. Our feature selection gives interesting results on this set. Indeed, using as few as 14 features, we are less than 1.9% behind the value obtained with all features for 10-fold cross-validation. Test values are following the exact same pattern.

Table 1. Results of the different classifiers for cross-validation and test

	LOO / 10-fold	Test	Comp. time
KNN 193	86.65%	85.89%	4.5min
KNN 193→14	93.20%	89.02%	60s
SVM 193	96.92%	96.76%	49h
SVM 193→14	95.08%	94.86%	4.5h

4.3 Analysis

From machine learning point of view, a major achievement was obtained: reducing the dimensionality by more than 13 and keeping roughly the same performance, in the variance interval. This result is interesting for different reasons: First, the computational time is drastically reduced, since the classifiers complexity relationships to dimensionality are linear. Second, because computational time is decreased by 11, it allows future new analyses and experiments previously not possible.

From steganalysis point of view, the obtained results are of course behind the actual best values, obtained for the Outguess algorithm in [12]. Nevertheless, the two advantages coming out of these results – namely the decrease of computational time and the gain in interpretability – can counterbalance this opinion. In the end, this set of features describes in a more precise way the functionning and problems of the Outguess algorithm. Taking these into account might help improve the steganographic scheme and make it less detectable: the first (and thus most "efficient") features selected by the forward algorithm show that the Outguess algorithm is especially weak when the analysis is made on features using -1 and -2 DCT coefficients, leading to already more than 90% of correct classification with the SVM classifier.

5 Conclusions and Future Work

This paper has presented a new methodology for dimensionality reduction by feature selection in the framework of steganalysis.

The issues of dimensionality have been adressed and the first step of our methodology proves that the theoretically required number of images for correct training is far from being needed. By the use of a Monte-Carlo technique on up to 4000 images, it has been shown that such numbers of images are sufficient for stable results. A set of 193 features extracted from all images serves the classification process, preceded by the dimensionality reduction step. This part of our methodology is achieved using a forward selection with a KNN classifier. It enables to reduce the number of required features to 14, while keeping roughly the same classification results. Computational time is thus greatly improved, divided by about 11. Further analysis becomes again possible with this low number of features: conclusions and precisions about the steganographic scheme can be infered from the obtained feature set. The last step using SVM for improvement over the previous KNN results achieves high classification results for so small a feature set, proving that many features among the full 193 set might not be relevant enough to be kept for classification purposes.

A comparison between the obtained reduced sets of features for various steganographic algorithms might reveal some common sensitive features. An analysis of these common points could help design a more generic steganalysis method using a "low" number of features.

Acknowledgements

Part of the work of Yoan Miche, Patrick Bas and Christian Jutten is supported by the French national funding under the project RIAM Estivale (ANR-05-RIAM-O1903). Part of the work of Yoan Miche, Olli Simula and Amaury Lendasse is supported by the project of New Information Processing Principles, 44886, of the Academy of Finland.

References

1. Devijver, P.A., Kittler, J.: Pattern recognition: a statistical approach. Prentice Hall, New York (1982)
2. Bellman, R.: Adaptive control processes: a guided tour. Princeton University Press, Princeton (1961)
3. François, D.: High-dimensional data analysis: optimal metrics and feature selection. PhD thesis, Université catholique de Louvain (September 2006)
4. McCullagh, D.: Secret messages come in .wavs. Online Newspaper: Wired News(February 2001) http://www.wired.com/news/politics/0,1283,41861,00.html
5. Boser, B. E., Guyon, I. M., Vapnik, V. N.: A training algorithm for optimal margin classifiers. In: Fifth Annual Workshop on Computational Learning Theory, pp. 144–152 (27-29 Juillet 1992)
6. Fridrich, J.: Feature-based steganalysis for jpeg images and its implications for future design of steganographic schemes. In: Information Hiding: 6th International Workshop, May 23-25, LNCS, vol. 3200, pp. 67–81. Springer, Heidelberg (2004)
7. Maxemchuck, J.M.: Electronic document distribution. AT and T Technical Journal 73(5), 73–80 (1994)
8. Verleysen, M., François, D.: The curse of dimensionality in data mining and time series prediction. In: IWANN'05 : 8th International Work-Conference on Artificial Neural Network, Lecture Notes in Computer Science, vol. 3512, pp. 758–770 (8-10 Juin, 2005)
9. Provos, N.: Defending against statistical steganalysis. In: 10th USENIX Security Symposium, pp. 323–335 (April 13-17, 2001)
10. Vapnik, V.N.: Statistical Learning Theory. Wiley-Interscience, New York (1998)
11. Christian, P.R., Casella, G.: Monte Carlo statistical methods. Springer, Heidelberg (1999) ISBN:038798707X.
12. Pevny, T., Fridrich, J.: Merging markov and dct features for multi-class jpeg steganalysis. In: IS and T/SPIE EI 2007, Lecture Notes in Computer Science, vol. 6505, January 29th - February 1st (2007)
13. Whitney, A.W.: A direct method of nonparametric measurement selection. IEEE Transactions on Computers C-20, 1100–1103 (1971)
14. Miche, Y., Roue, B., Lendasse, A., Bas, P.: A feature selection methodology for steganalysis. In: Gunsel, B., Jain, A.K., Tekalp, A.M., Sankur, B. (eds.) MRCS 2006. LNCS, vol. 4105, pp. 49–56. Springer, Heidelberg (2006)