Co-occurrence steganalysis in high dimensions

Tomáš Pevný

Agent Technology Center, Czech Technical University in Prague,
Karlovo naměstí 13, 121 35 Prague 2, Czech Republic.

ABSTRACT

The state of the art steganalytic features for spatial domain, and to some extent for transfer domains (DCT) as well, are based on histogram of co-occurrences of neighboring elements. The rationale behind is that neighboring pixels in digital images are correlated, which is caused by the smoothness of our world and by the usual image processing. The limitation of the histogram-based features is that they do not scale well with respect to the number of modeled neighboring elements, since the number of histogram bins (hence number of features) depends exponentially on this quantity.

The remedy adopted by the prior art is to sum values of neighboring bins together, which can be seen as a vector quantization controlled by the position of the quantization centers. So far the quantization centers has been determined manually according to the intuition of the steganalyst. Here we propose to use Linde, Buso, and Gray algorithm in order to automatically find quantization centers maximizing the detection accuracy of resulting features. The quantization centers found by the proposed algorithm are experimentally compared to the ones used by the prior art on the steganalysis of Hugo algorithm. The results show a non-negligible improvements in the accuracy, especially when more complicated filters and higher-order histograms are used.

1. INTRODUCTION

The state of the art steganalytic features, especially for spatial domain, rely on histogram of co-occurrences. The potential of this approach has been probably first investigated in[17, 21] by using Markov models. Albeit the reported results were promising, almost no progress has been done until the introduction of SPAM features[14] designed to break the LSB matching algorithm. Later on, SPAM features have shown to be efficient in breaking YASS algorithm,[19] especially after they have been augmented by Cartesian-Calibrated PF274 features.[10] The histogram of co-occurrences based features has been further improved during BOSS contest introduced to break HUGO algorithm.[15] The "Hugobreakers" team created steganalyzer, which extracted co-occurrence features of 3 and 4 elements from image residues calculated by using large number of filters.[7, 11] Their concept called "rich models" has proven to be very efficient for steganalysis in spatial and transformed domain as well.

The above short summary shows that the contemporary steganalysis relies on histogram of co-occurrences. This promising approach is limited by number of histogram bins, which depends on the dimension of the histogram (number of modeled co-occurring elements), and complexity of the image filter (use to improve signal to noise ratio). The number of histogram bins bins grows exponentially with respect to histogram's dimension and linearly with respect to the complexity of the filter.

The prior art used two approaches to decrease number of histogram bins. While the first approach used subset of histogram bins directly as steganalytic features,[14, 20] the second combined neighboring histogram bins together.[7, 17] From the machine learning point of view, the first approach can be seen as a form of feature selection, the second one as a form of feature extraction process. In both cases, the bins were either selected or combined by hand without any optimization.

From results of the prior art we can conclude that combing neighboring histogram bins together is better then selecting their subset, thus this approach is investigated in this paper. We argue the this problem is in its essence a vector quantization problem allowing us to convert the feature extraction problem to the problem of designing a vector quantizer (i.e. search for optimal quantization bins). For this problem an efficient algorithm exist,[2] which albeit not being designed for feature extraction, can be used due to unique properties of digital images. We also proposes a variant of the algorithm designed to highlight differences in probability distributions of cover and stego images.

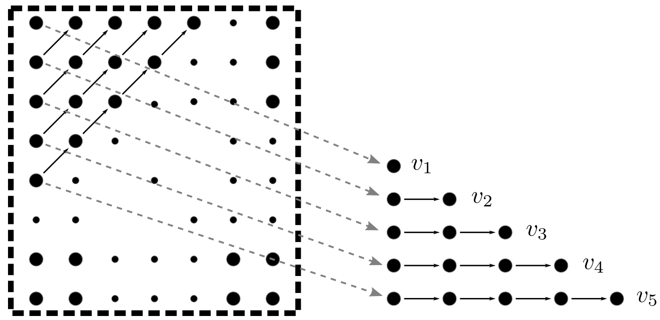


Figure 1: Schematics of the permutation (step 2) operating along minor diagonal line.

The proposed algorithm(s) are not designed as a replacement of general feature selection / extraction algorithms. Their aim is to optimize the position of quantization centers to maximize the accuracy of the steganalysis. This should simplify the use of more complicated filters and it also enables to use more sophisticated machine learning algorithms, which cannot deal with a huge dimension of unquantized histograms,

In the experimental section, the proposed algorithms are used to find quantization centers for histograms of co-occurrences of 3 and 4 elements and for filters of orders 1,. . . ,6[7] (see Table 1) to detect the Hugo algorithm.[15] The founded quantization centers were compared to the quantization centers used in the prior art (determined by steganalyst's intuition). The experimental results show that the quantization centers found by the proposed solution are almost always better then the ones used in the prior art.

This paper is organized as follows. The next section formalizes the feature extraction process followed by vast majority of contemporary features. Section 3 introduces the problem of histogram quantization, for which heuristic solutions are proposed in Section 4. Section 5 contains the experiments. Finally, the paper is concluded in Section 6.

2. FORMAL MODEL OF FEATURE EXTRACTION

It is rather surprising that the calculation of most state of the art steganalytic features* follows the similar pattern, which can be broken down into four steps: (a) filtering an image with a filter, (b) permuting filtered elements, (c) calculating histogram, and (d) down-sampling of a histogram. Also this scheme has been already used in the prior art, Gül et al.[8] describes these steps explicitly with the goal to optimize several parameters of this process to improve the sensitivity of features to detect particular steganographic algorithm. The rationale behind the steps is following:

1. Image filtering is used to increase the signal-to-noise ratio, where the signal is steganographic changes introduced by the embedding and the noise is the image content (i.e. Eve wants to suppress the image content). The simplest filter is the edge detector $\begin{bmatrix} +1 & -1 \end{bmatrix}$, but more sophisticated filters were already used in the prior art.[9,11,12] Here, we would like to point out that the DCT, Wavelet, and other transformations involved in calculation of some features sets can be seen as a simultaneous application of multiple filters.

2. The permutation is introduced mainly to simplify the notation of the calculation of the histogram of co-occurrences (following steps), which are usually calculated along different paths. With the help of the permutation, we can write the calculation of histograms as an operation on sets of vectors, $\mathbf{V} = \{v_i \in \mathbb{R}^{m_i} | i \in \{1, \ldots, l\}\}$, of variable length. The most popular permutations order elements such that the histograms are calculated along horizontal, vertical, minor and major diagonal (Shi et al.[17]). Other scans certainly exists, e.g. scan along the Hilbert curve, zig-zag, etc. An example of the permutation operating along minor diagonal is in Figure 1.

*By blind we mean that the steganalytic features can detect more than one algorithm.

3. With the help of the permutation introduced in the previous step, histogram of co-occurrences of t elements extracted from an image x is calculated as

$$\mathcal{F}_j(x) = \frac{1}{Z}\sum_{i=1}^{l}\sum_{k=1}^{m_i-t} I_{d_j}(v_{i,k},\ldots,v_{i,k+t}), \tag{1}$$

where $Z = \sum_{i=1}^{l} m_i - t$ is the normalization, $d \in \mathbb{R}^t$ denotes the bin of the histogram, and $I_d(v_{i,k},\ldots,v_{i,k+t})$ is the indicator function

$$I_{d_j}(v_{i,k},\ldots,v_{i,k+t}) = \begin{cases} 1 & \text{if } (v_{i,k},\ldots,v_{i,k+t}) = d_j, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

The set of all histogram bins is denoted as $\mathbf{D} = \{d_i \in \mathbb{R}^t\}_{i=1}^{s}$.

4. Finally, the histogram is down-sampled to produce final set of features. The down-sampling is necessary, because the number of histogram bins grows exponentially with the number of modeled co-occurrences, and due to the curse of dimension, bins cannot be directly used as features. The most common form of down-sampling is to omit less populated histogram bins (e.g. SPAM features[14]) or to add less populated bins to more populated bins (e.g Markov features[17]), and using symmetries.[12]

Although the above list of steps is not exhaustive (features can be for example use calibration[6]), they are in some form present in the extraction mechanism of almost all steganalytic features.

Let's now imagine that Eve knows the steganographic algorithm Alice and Bob uses and she wants to optimize every step in the feature extraction process in order to maximize the sensitivity of her detector (features). The simultaneous optimization would be probably difficult, computationally expensive, and there will be a danger of over-fitting. As a remedy to these issues, she performs the optimization of individual steps independently.

The focus of this paper is on the fourth step, namely on the quantization. The goal is to find function projecting high-dimensional histogram (imagine histogram of co-occurrences of four, five, etc. elements) to its quantized version such that the loss of information important for steganalysis is minimized.

3. HISTOGRAM QUANTIZATION

Let start by introducing the needed notation related to histograms. Throughout the paper it is assumed that the filter in step 1, permutation in step 2, and the number of modeled elements (we call it dimension of the histogram), t, in step 3 are fixed and known beforehand. By virtue of these assumptions, the set of all histograms bins, $\mathbf{D} = \{d_i \in \mathbb{R}^t\}_{i=1}^{s}$, in (2) is finite. Given an image, x, steps 1–3 in the feature extraction process (this means filtering, permutation, and the calculation of the histogram) can be viewed as a projection, $\mathcal{F} : \mathcal{X} \mapsto \mathbb{R}^s$, from the space of all images, $\mathcal{X}$, to the Euclidean space $\mathbb{R}^s$, where s denotes the number of histogram bins. s is determined by the number of colors in digital images, chosen filter, and dimension of the histogram. In practice, s can be in order of tens of millions and therefore the number of histogram bins needs to be reduced, which will be done in this case by the quantization.

3.1 Quantization

Let $\mathbf{Q} = \{q_i \in \mathbb{R}^t | i \in \{1,\ldots,n_q\}\}$ denotes the set of quantization points in $\mathbb{R}^t$. Then, by quantization of $\mathcal{F}(x)$ to $\mathbf{Q}$, we understand $\mathcal{Q} = \{\mathcal{Q}(q_i)\}_{i=1}^{d}$ such that

$$\mathcal{Q}(q_i) = \sum_{d_j \in \mathrm{NN}(q_i)} \mathcal{F}_j(x),\ i \in \{1,\ldots,n_q\},$$

where $\mathrm{NN}(q_i)$ denotes the set of nearest neighbors of q_i, i.e.

$$\mathrm{NN}(q_i) = \{d_j \in \mathbf{D}\,|(\forall k)\,(\|d_j - q_i\| \leq \|d_j - q_k\|)\}\,.$$

Since the histogram (1) in step 3 is not quantized, it can be assumed it carries all the information potentially useful for steganalysis (for a given combination of filter, permutation, and dimension of the histogram). By its quantization to $\mathbf{Q}$, some of the information is lost and the goal is to find quantization centers $\mathbf{Q}$ such that this loss would be minimal.

The quantization together with the projection $\mathcal{F}$ and probability distribution functions (pdfs) of cover, p_C, and stego objects, p_S, on $\mathcal{X}$ implicitly defines pdfs $p_C^{\mathbf{D}}$ and $p_S^{\mathbf{D}}$, on the quantized space $\mathbb{R}^{n_q}$. The goal of minimizing the information loss during quantization can be stated as a maximization of KL-divergence

$$\max_{q_1,\ldots,q_d} D_{\mathrm{KL}}(p_c^{\mathbf{D}} \| p_s^{\mathbf{D}}) = \int_{\mathbb{R}^d} p_C^{\mathbf{D}}(x) \log p_C^{\mathbf{D}}(x) dx - \int_{\mathbb{R}^d} p_C^{\mathbf{D}}(x) \log p_S^{\mathbf{D}}(x) dx. \tag{3}$$

The direct optimization of (3) is impossible from many reasons: (a) the dimension of the optimization problem is very high ($d \times t$), (b) the optimization problem is not continuous due to the nearest-neighbor operation, (c) the pdfs p_c and p_s on $\mathcal{X}$ are not known, hence all the quantized pdfs $p_C^{\mathbf{D}}$ and $p_S^{\mathbf{D}}$ are not known either.

The lack of knowledge about p_c and p_s is usually resolved by estimating the quantity using set of samples. In this case, the KL-divergence can be estimated by using the KNN estimator.[4,18] Let $\mathbf{X} = \{x_i\}_{i=1}^{D}$ and $\mathbf{Y} = \{y_i\}_{i=1}^{D}$ denote set of D cover, and D stego images respectively. Let $\rho_k(\mathbf{X}, z)$ and $\rho_k(\mathbf{Y}, z)$ denote the radius of the smallest ball in the quantized space $\mathbb{R}^{n_q}$ containing exactly k samples from $\mathbf{X}$ and $\mathbf{Y}$, respectively. Then,

$$\hat{D}_{\mathrm{KL}}(p_c^{\mathbf{D}} \| p_s^{\mathbf{D}}) = \log \frac{D}{D-1} + \frac{n_q}{D} \left(\sum_{i=1}^{D} \log \rho_k(\mathbf{X}, x_i) - \sum_{i=1}^{D} \log \rho_k(\mathbf{Y}, x_i) \right) \tag{4}$$

is a consistent and asymptotically unbiased estimator of the KL divergence as long as $k/D \to 0$, $k \geq n_q$, and $k \to \infty$ as $D \to \infty$. For large D, the first term is approximately zero. The second and the third terms are estimates of the entropy $H(p_c^{\mathbf{D}})$ and the cross-entropy $H_{\times}(p_c^{\mathbf{D}}, p_s^{\mathbf{D}})$.

The accuracy of KNN the estimator (4) has been experimentally investigated in steganographic settings[16] with the conclusion that its accuracy is not sufficient to verify the security of steganographic scheme. But (4) provide us with the intuition for the optimization problem (3) that the quantization centers $\mathbf{Q}$ should be chosen such that distances between feature vectors extracted from cover images would be small and distances between feature vectors extracted from cover and stego images would be large.

The optimization criteria based on the above thoughts can be formalized as

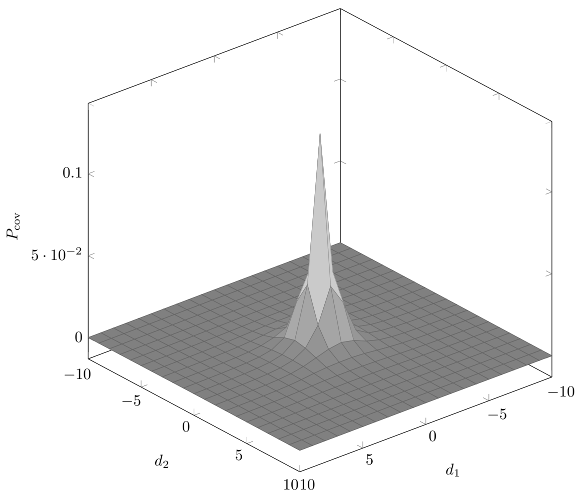
$$\max_{q_1,\ldots,q_d} \sum_{i=1}^{d} \left[\sum_{c_j \in \mathrm{NN}(q_i)} \left(\sum_{x \in \mathbf{X}} \mathcal{F}_j(x) - \sum_{y \in \mathbf{Y}} \mathcal{F}_j(y) \right) \right]^2. \tag{5}$$

It maximizes the L_2 distance between average cover and stego feature vectors estimated from a set of samples. Although the optimization problem (5) is oversimplified (e.g. it does not take into account correlation between features), it is still difficult to be solved due to high dimension and the optimization function not being continuous. Consequently, we resort to heuristics introduced in the next section.

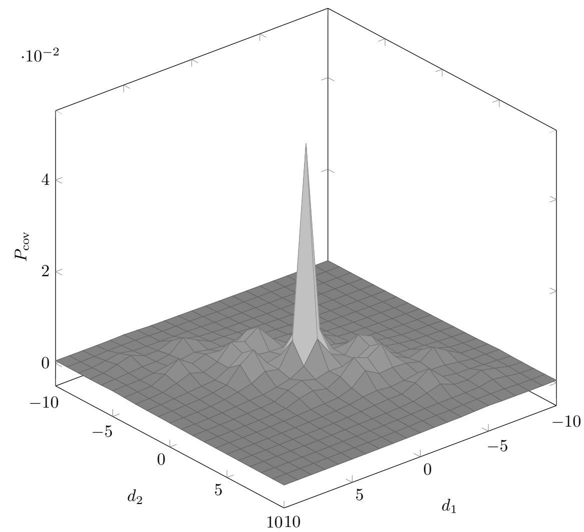
4. HEURISTIC SOLUTIONS

Histograms of filtered digital images† have some unique properties, which are exploited by the proposed heuristic approaches to the quantization problem described in the previous section. The histograms usually have large number of bins, but only few of them are populated enough to present a robust features for steganalysis. Values of many bins extracted from cover and from the same stego images are essentially the same, making them useless for steganalysis. Third, the histograms are smooth in the sense that bins close together often have similar values.

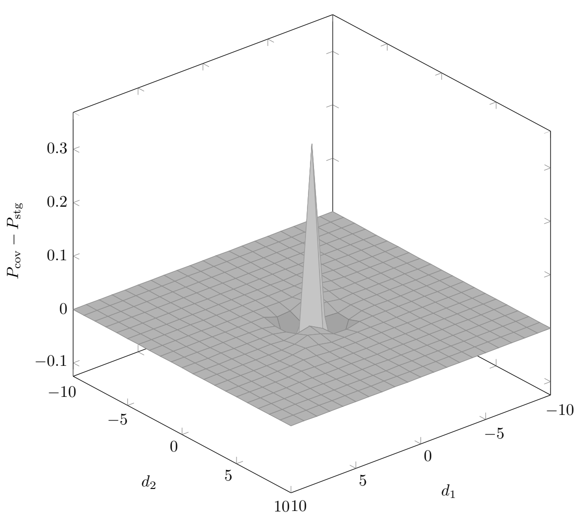
†Here, we assume filters suppressing image content and enhancing steganographic noise.



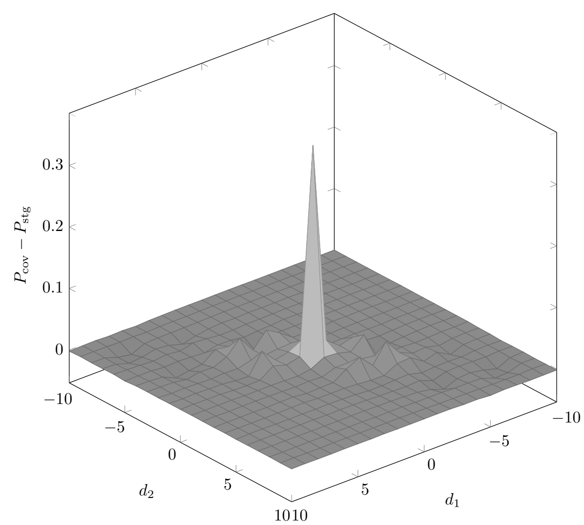
(a) p_c, filter $[+1, -1]$



(b) p_c, filter $[+1, -4, +6, -4, -1]$



(c) $p_c - p_s$, filter $[+1, -1]$



(d) $p_c - p_s$, filter$[+1, -4, +6, -4, +1]$

Figure 2: (a), (b) Average probability of two neighboring residues ($t = 2$) in 100 cover images. (c), (d) normalized difference of average probability of two neighboring residues ($t = 2$) in cover and stego images. All probabilities were estimated from residues using filters $[+1, -1]$ and $[+1, -4, +6, -4, -1]$. Histograms were estimated from 100 pairs of cover and the same stego images embedded by Hugo[15] algorithm with payload $\alpha = 0.4$ and $T = 80$.

Figures 2(a,b) shows the probability of two neighboring residues ($t = 2$) in cover images by using filters $[+1, -1]$ and $[+1, -4, +6, -4, -1]^{\ddagger}$ and calculated along horizontal and vertical direction. We can see that most of the information about covers is located in the small area around center point $[0, 0]$. The probability very quickly decreases with the distance from the origin. Similar behavior was observed for filters $[+1, -2, +1]$ and $[+1, -3, +3, -1]$.

Figures 2(c,d) shows the *difference* in probability of two neighboring residues ($t = 2$) in cover and stego images calculated under the same conditions. For both filters, we can observe a sharp peak in zero with minimums and maximums located along major diagonal. Again, the similar behavior was observer for filters $[+1, -2, +1]$, and $[+1, -3, +3, -1]$.

In both cases, most of the information (about cover images and about the difference between cover and stego images) are located in the small region around zero and along the major diagonal. It is surprising that almost no information is located along minor diagonal, but this might be a unique property of the chosen filters.

Recall now our goal, which is to find n_q quantization centers, such that they well characterizes the differences between probability distributions in cover and stego images. With the respect to the above, these bins should be located (a) in areas with high probability and (b) in areas with high differences between cover and stego objects. In the rest, we propose two approaches toward the goal.

4.1 Modeling the cover distribution

Let's simplify the problem by focusing on the best characterization of cover distribution for a given number of quantization centers n_q. This problem can be viewed as a quantization problem, where we want to minimize squared-error distortion caused by the quantization. Solution to this problem has been proposed by Linde, Buzo, and Gray in 80's in Ref. 2. By using the notation introduced above, the algorithm (called LBG), is designed to minimize

$$\sum_{i=1}^{n_q} \sum_{d_j \in \mathrm{NN}(q_i)} \mathcal{F}_j(x) \|d_j - q_i\|^2, \tag{6}$$

where $\mathcal{F}_j(x)$ denotes the value (probability) of the bin d_j in the full histogram extracted from image x and q_i denotes the quantization center $q_i \in \mathbf{Q}$.

Notice that the criteria (6) utilizes only (average) histogram of cover or stego images. Thus, the LBG algorithm finds quantization centers characterizing probability distribution in either cover or stego images, but not highlighting the difference between them.

Comparing Figures 5(a,b) with Figures 5(c,d), we can observe that the information about covers, and about differences between covers and stegos are located in the same regions. This phenomenon is caused by the fact that bins with higher probability are more likely to be modified by the embedding than the bins with low probability.

The LBG algorithm used to find quantization centers to model probability distribution in cover images is well suited for universal steganalysis. But with the respect of the above, the same quantization centers can be used for targeted steganalysis as well with very good results.

We point out that the Hugo algorithm tries to preserve model capturing co-occurrences of three residues with filter $[+1, -1]$. Figures 5(a,c) shows this situation with the difference of capturing only two elements. Yet we can see that the highest differences are located in the areas of high probability.

In this paper, we call this approach modeling distributions in cover images *lbg* algorithm. Besides the lbg algorithm, we also evaluate the simple approach taking n_q bins with the highest estimated probability. We call this approach *top*.

‡The histograms are estimated from 100 pairs of cover image and from the same stego image embedded by Hugo[15] algorithm with payload $\alpha = 0.4$ and $T = 80$. The images were randomly chosen from Bossbase-1.0.

order	normalization	filter
1	1	$[+1, -1]$
2	2	$[+1, -2, +1]$
3	3	$[+1, -3, +3, -1]$
4	6	$[+1, -4, +6, -4, +1]$
5	10	$[+1, -5, +10, +5, -1]$
6	20	$[+1, -6, +15, -20, +15, -6, +1]$

Table 1: Filters to calculate image residues used in the experiments.

4.2 Modeling the differences between distributions

To tweak the LBG algorithm to characterize regions with differences between cover and stego distributions, the LBG algorithm is executed twice. In the first execution, the LBG algorithm finds $\frac{n_q}{2}$ quantization centers approximating the probability distribution of all bins d_j with a *positive* difference $\mathcal{F}_j(x^{\text{cov}}) - \mathcal{F}_j(x^{\text{stg}}) > 0$. In the second execution, the LBG algorithm finds remaining $\frac{n_q}{2}$ quantization centers approximating the probability distribution of all bins d_j with a *negative* difference $\mathcal{F}_j(x^{\text{cov}}) - \mathcal{F}_j(x^{\text{stg}}) < 0$. Finally, the quantization centers found in both executions are joined together and the joined set $\mathbf{Q}$ is used as a solution. The solution of the presented heuristic algorithm is certainly not optimal, but the belief is that it is reasonably good, since the LBG algorithm used to solve the sub-problems is asymptotically optimal.

This algorithm is called *lbg-diff.* As in the previous section, we also evaluate the simple approach taking n_q bins with the highest differences in probability of cover and stego images. We call this approach *diff-top.*

4.3 Implementation details

The biggest obstacle in the practical realization of the presented heuristic algorithms is the implementation of the non-quantized histogram $\mathcal{F}(x)$, because the number of its bins can be very large. In order to keep the number of its bins reasonable low, we have restricted the algorithm to use only bins with the probability (of difference) higher than the design threshold τ. In experimental section it will be shown that this only parameter of the algorithm (besides the number of quantization bins n_q) plays an important role. It controls the extent, to which the algorithm approximates bins with very low probability, which greatly contribute to the distortion (6) if they are far from the quantization center. In our experiments, we have tried four different values of $\tau \in \{0, 10^{-5}, 10^{-4}, 10^{-3}\}$. In the case of $\tau = 0$, we have used 10^7 most important histogram bins to make the algorithms computationally feasible.

5. EXPERIMENTS ON HUGO

This section presents empirical evaluation of the proposed methods for finding quantization centers. Namely methods *top, diff-top, lbg,* and *lbg-diff* are compared. Since the last two methods depend on the parameter τ, four different values of $\tau \in \{0, 10^{-5}, 10^{-4}, 10^{-3}\}$ were compared. Thus, for every number of quantization centers n_d, number of modeled co-occurrences $t \in \{3, 4\}$, and type of the filter (Table 1) , we tested 10 different methods.

The methods were tested on six different filters (Table 1) introduced in.[11] Contrary to the Ref. 11, the image residues after filtering were not divided by the normalization and they were not rounded. These operations are considered as a part of the quantization, which should be determined by the presented methodology.

The quantization centers founded by the proposed heuristics have been compared to quantization centers used in the prior art,[11,14] which are

$$\mathbf{Q}^{3d} = \{(i, j, k) \,| i, j, k \in \{-T, \dots, T\}\} \tag{7}$$

for histograms of co-occurrences of 3 elements (we use $T = 3$ to obtain 343 quantization centers), and

$$\mathbf{Q}^{4d} = \{(i, j, k, l) \,| i, j, k, l \in \{-T, \dots, T\}\} \tag{8}$$

for histograms of co-occurrences of 4 elements[7] (we use $T = 2$ to obtain 625 quantization centers). In the prior art the image residues are divided by the normalization constant (see Table 1) and rounded to the nearest integer.

		τ	256	343	512	1024
cover	top	—	**0.388**	0.383	0.387	**0.370**
	lbg	10^{-3}	0.409	0.396	0.396	0.391
	lbg	10^{-4}	0.415	0.413	0.390	0.382
	lbg	10^{-5}	0.429	0.423	0.422	0.400
	lbg	0	0.446	0.452	0.433	0.411
cover - stego	diff	—	0.396	0.387	**0.381**	0.372
	lbg-diff	10^{-3}	—	—	—	—
	lbg-diff	10^{-4}	0.412	0.393	0.387	—
	lbg-diff	10^{-5}	0.442	0.442	0.424	0.412
	lbg-diff	0	0.478	0.483	0.473	0.456
reference - hypercube			—	**0.381**	—	—

Table 2: Average classification error for the proposed heuristics and the prior art. The steganalytic features were based on histogram of co-occurrences of 3 elements and filter $[+1, -1]$. Methods utilizing only cover images, and cover and stego images are separated. Captions $256, 343, 512$, and 1024 denote number of quantization centers.

This operation is equivalent to the multiplication of quantization centers by the appropriate normalization constant. In our experiments, we have followed the latter approach. In this paper, these quantization centers are called *hypercube*.

Steganalytic features were extracted following the methodology introduced in Section 2. Histograms of co-occurrences were calculated along horizontal and vertical axis. Corresponding bins from both histograms were summed exactly in the same way as in Ref. 11, 14 and normalized such that the sum of the resulting feature vector was equal to one.

The quality of founded quantization centers (features) has been estimated by the ensemble classifier recently proposed in Ref. 13 with training/testing ratio set to 0.33 and with automatic setting of number of weak classifiers and dimension of randomly selected features. We report average error $P_{\mathrm{e}} = \frac{1}{2}(P_{\mathrm{fp}} + P_{\mathrm{fn}})$ from 10 repetitions of the training and testing with different splits (P_{fp}/ P_{fn} corresponds to estimated probability of false positive / false negative respectively).

Finally, the experiments utilized Boss-base 1.0[3] containing 10 000 images. The material section of BOSS site[5] offers images, which are already scaled and cropped to fixed size of 512×512 and converted to greyscale. The same images are available after being embedded by Hugo[15] algorithm with a fixed payload $\alpha = 0.4$ and with $T = 80$. These small images were used in the experiments.

5.1 Co-occurrences of 3 elements

This subsection discusses results obtained by using histogram of co-occurrences of 3 elements. Results obtained by using filters $[+1, -1]$ and $[+1, -4, +6, -4, +1]$ are discussed in more detail, because they exhibit an opposite pattern. The results on other filters are similar, and consequently, they are discussed briefly.

Results on features extracted with the filter $[+1, -1]$ (Table 2) show that the best error for 343 quantization centers has been achieved by using *hypercube* (7) approach, but quantization centers determined by *top* and *diff* approaches closely follows. In fact, differences between errors are very small and they might not be statistically significant. *Top* and *diff* approaches do not utilize the lbg algorithm and we believe that the reason for their superiority is that for this filter the number of bins with high probability is very small (i.e. the probability distribution is concentrated around the origin) and merging neighboring bins together is not an advantage.

As has been discussed above, the LBG algorithm approximates whole probability distribution. This means that centers with small probability but far away from the nearest quantization center can have large contribution to the distortion function (6) and therefore they greatly influence the final solution, despite not being important for the classification problem. The fact that quantization centers found by *lbg* and *lbg-diff* algorithm with high values of $\tau = 10^{-3}/10^{-4}$ provide errors close to those of *top* and *diff* supports this explanation, as in these cases the algorithm did not utilize bins with small probability.

		τ	256	343	512	1024
cover	top	—	0.325	0.318	0.314	0.303
cover	lbg	10^{-3}	0.315	0.313	0.303	0.296
cover	lbg	10^{-4}	**0.307**	**0.303**	0.302	0.292
cover	lbg	10^{-5}	0.320	0.312	**0.299**	**0.292**
cover	lbg	0	0.367	0.366	0.348	0.326
cover - stego	diff	—	0.327	0.320	0.311	0.298
cover - stego	lbg-diff	10^{-3}	—	—	—	—
cover - stego	lbg-diff	10^{-4}	0.317	0.307	0.307	0.295
cover - stego	lbg-diff	10^{-5}	0.329	0.319	0.306	0.303
cover - stego	lbg-diff	0	0.436	0.432	0.421	0.389
hypercube			—	0.322	—	—

Table 3: Average classification error for the proposed heuristics and the prior art. The steganalytic features were based on histogram of co-occurrences of 3 elements and filter $[+1, -4, +6, -4, +1]$. Methods utilizing only cover images, and cover and stego images are separated. Captions $256, 344, 512$, and 1024 denote number of quantization centers.

order of filter	*lbg*	*lbg-diff*	*hypercube*
1	0.396 ($\tau = 10^{-3}$)	0.393 ($\tau = 10^{-4}$)	0.381
2	0.316 ($\tau = 10^{-3}$)	0.319 ($\tau = 10^{-4}$)	0.314
3	0.308 ($\tau = 10^{-4}$)	0.308 ($\tau = 10^{-4}$)	0.335
4	0.303 ($\tau = 10^{-4}$)	0.307 ($\tau = 10^{-4}$)	0.322
5	0.305 ($\tau = 10^{-5}$)	0.305 ($\tau = 10^{-5}$)	0.327
6	0.302 ($\tau = 10^{-5}$)	0.308 ($\tau = 10^{-5}$)	0.324

Table 4: The average classification error for variants of *lbg* and *lbg-diff* heuristics providing the lowest values for 343 quantization centers. The average error on *hypercube* centers is provided for reference. The steganalytic features were based on histogram of co-occurrences of 3 elements and filters from Table 1.

On the other hand, results on features extracted with the filter $[+1, -4, +6, -4, +1]$ (Table 3) show that the proposed *lbg* heuristic algorithm found quantization centers superior to the *hypercube*. Moreover, the *top* and *diff* approaches are inferior to those utilizing LBG algorithm. These results supports the conclusions of the previous paragraph. The probability distribution in case of more complicated filters is not peaky, but rather flat and there is a large number of bins with high probability (see Figures 2(a,b)). The LBG algorithm quantizes many bins together, which better characterizes the distribution for a given number of quantization centers, and leads to the better classification.

Table 4 shows average errors achieved by using 343 quantization centers returned by *lbg* and *lbg-diff* heuristics. The errors are provided for six different filters together with the average error on *hypercube* centers. We can observe that the *hypercube* quantization centers are better only for the first order filter. For the third and higher order filters the quantization centers returned by the proposed heuristics are better by two or more percents.

5.2 Co-occurrences of 4 elements

The average errors of features modeling co-occurrences of four elements using the simplest filter $[+1, -1]$ is shown in Table 5. The results support the conclusions drawn in the previous subsection. The probability distribution of co-occurrences of four residues is not as peaky as in the case of three residues. This phenomenon is exhibited by the fact that the *hypercube* quantization centers provides similar error as the quantization centers found by lbg based heuristics.

Table 6 shows average errors achieved by using 625 quantization centers returned by *lbg* and *lbg-diff* algorithms together with errors on *hypercube* quantization centers. Again, the errors are shown for all six filters from Table 1. We can observe that quantization centers found by the proposed algorithms are better (the errors are on average by two percents lower) that the hand-designed ones. Also notice that the improvement diminishes with the

		τ	256	512	625	1024
cover	top	—	**0.307**	**0.291**	**0.288**	0.286
	lbg	10^{-3}	0.315	0.292	0.297	0.285
	lbg	10^{-4}	0.320	0.296	**0.288**	0.281
	lbg	10^{-5}	0.343	0.304	0.303	0.283
	lbg	0	0.460	0.422	0.412	0.376
cover - stego	diff	—	0.332	0.302	0.294	**0.280**
	lbg-diff	10^{-3}	—	—	—	—
	lbg-diff	10^{-4}	0.314	—	—	—
	lbg-diff	10^{-5}	0.338	0.304	0.300	0.284
	lbg-diff	0	0.478	0.462	0.454	0.444
reference - hypercube			—	—	**0.288**	—

Table 5: Average classification error for the proposed heuristics and the prior art. The steganalytic features were based on histogram of co-occurrences of 4 elements and filter $[+1, -1]$. Methods utilizing only cover images, and cover and stego images are separated. Captions $256, 512, 625$, and 1024 denote number of quantization centers.

order of filter	*lbg*	*lbg-diff*	*hypercube*
1	0.288 ($\tau = 10^{-4}$)	0.289 ($\tau = 10^{-4}$)	0.288
2	0.265 ($\tau = 10^{-5}$)	0.266 ($\tau = 10^{-5}$)	0.279
3	0.263 ($\tau = 10^{-5}$)	0.266 ($\tau = 10^{-5}$)	0.284
4	0.283 ($\tau = 10^{-4}$)	0.289 ($\tau = 10^{-5}$)	0.302
5	0.293 ($\tau = 10^{-5}$)	0.303($\tau = 10^{-5}$)	0.313
6	0.306 ($\tau = 10^{-5}$)	0.306 ($\tau = 10^{-5}$)	0.317

Table 6: The average classification error for variants of *lbg* and *lbg-diff* heuristics providing the lowest values for 625 quantization centers. The average error on *hypercube* centers is provided for reference. The steganalytic features were based on histogram of co-occurrences of 4 elements and filters from Table 1.

higher order. We believe that this is because after filtering by higher order filters, there would be relatively few histogram bins with the probability above the threshold τ and it should be therefore lowered.

5.3 General discussion

By careful comparison of the best error rates achieved on quantization centers found by *lbg* and *lbg-diff* approaches (Tables 4 and 6), one can see that they are very similar. This phenomenon is not intuitive, as one would guess that *lbg-diff* approach better enhances differences between probability distributions of cover and stego images leading to better solution. But the experimental results does not support this intuition. We believe that this is because areas with highest difference in probability coincide with areas with high probability (see Figure 2) and unless the steganographic algorithm is not particularly flawed, the embedding changes occur at areas with high probability, which are modeled by the LBG algorithm. This is an important finding for universal steganalysis, as it shows that *lbg* algorithm can be used to find good quantization centers. In the same time, there are cases when the *lbg-diff* approach would be better, for example the case of weakness in Hugo algorithm described in.[7]

Results in Tables 4 and 6 show that the optimal value τ of the threshold on histogram bins used in the LBG algorithm depends on how peaky is the probability distribution function to be characterized and on the number of quantization centers to be founded. The general guideline is following. For small number of quantization centers and peaky probability distributions its value should be higher. For higher number of quantization centers and flat probability distributions its value should be smaller. Since the dependency between τ and the error rate is straightforward, we believe that it would be possible to create algorithm to find an optimal value.

6. CONCLUSION AND FUTURE WORK

This paper presents a method to improve the current art on the histograms down-sampling in co-occurrence based steganalytic features by merging neighboring bins together. It has argued, why the problem is similar to the problem of designing a vector quantizer. Since the latter problem has been already successfully solved by the prior art, we have adopted the solution to our problem. We have also proposed a modification in order to highlight differences between distributions in cover and stego images.

The presented approach to the reduction of histogram bins has been experimentally compared to the solution used in the prior art. In the experiments, we have attacked the Hugo algorithm which represents the state of the art steganographic algorithm for spatial domain. The experiments showed that the proposed approach non-negligibly increases the quality of the resulting features, hence the accuracy of the steganalysis. It also significantly simplifies the use of more complicated filters and paves the road towards features based on histogram of co-occurrences of five elements. The last statements needs to verified, which is the plan for the future work.

We also highlight that the algorithm utilizing only probability distributions in cover images achieved performance similar to the algorithm utilizing probability distributions in cover and stego images. The explanation of this phenomenon is that histogram bins with high probability in cover images are affected more by the embedding operation that the ones with the low probability. Therefore areas of high probability coincides with areas of high difference in probability, which makes the solutions to be similar. We believe this results to be important for the universal steganalysis.

7. ACKNOWLEDGMENTS

The work on this paper was supported by European Office of Aerospace Research and Development under the research grant number FA8655-11-3035. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation there on. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of EOARD or the U.S. Government.

APPENDIX A. LBG ALGORITHM

Let assume we have an set of histogram bins $\mathbf{D} = \{d_i \in \mathbb{R}^t\}_{i=1}^s$ with corresponding probabilities (frequencies) $\{f_i \in \mathbb{R}\}_{i=1}^s$. The goal is to find a set of quantization centers $\mathbf{Q} = \{q_i \in \mathbb{R}^t | i \in \{1, \ldots, n_d\}\}$ minimizing the objective function

$$\sum_{i=1}^{n_d} \sum_{d_j \in \mathrm{NN}(q_i)} f_j \|d_j - q_i\|^2.$$

LBG algorithm is an iterative algorithm, which starts with one quantization center. In every iteration, all centers are split into two new centers. This process is repeated until the desired number of quantization centers is obtained. The following description of the algorithm has been taken from.[1]

Initialization: The initial quantization center is a simple weighted average of all bins,

$$q_1^* = \frac{\sum_{j=1}^s f_j d_j}{\sum_{j=1}^s f_j}.$$

For the distortion used in the stopping criteria holds

$$D^* = \sum_{j=1}^s f_j \|d_j - q_1^*\|^2.$$

Splitting: Every quantization center is split into to. Assuming there is n_d quantization centers, the new centers are created as follows

$$\begin{aligned} q_i^{(0)} &= (1-\epsilon) q_i^* \\ q_{i+n_d}^{(0)} &= (1+\epsilon) q_i^*, \end{aligned}$$

where ϵ is a small positive constant and $i \in \{1, \ldots, n_d\}$. After all quantization centers are split, we set $n_d = 2n_d$.

Optimization of centers: Set $k = 0$ and $D^{(0)} = D^*$

1. Update the quantization centers
$$q_i^{(k+1)} = \frac{\sum_{d_j \in \mathrm{NN}(q_i^{(k)})} f_j d_j}{\sum_{d_j \in \mathrm{NN}(q_i^{(k)})} f_j},$$
where $i \in \{1, \ldots, n_d\}$ and $\mathrm{NN}(q_i^{(k)})$ denotes the set of nearest neighbors of $q_i^{(k)}$, i.e.
$$\mathrm{NN}(q_i^{(k)}) = \left\{ d_j \in \mathbf{D} \middle| (\forall r) \left(\|d_j - q_i^{(k)}\| \leq \|d_j - q_r^{(k)}\| \right) \right\}.$$

2. Set $k = k + 1$.

3. Update the distortion
$$D^{(k)} = \sum_{i=1}^{n_d} \sum_{d_j \in \mathrm{NN}(q_i^{(k)})} f_j \|d_j - q_i^{(k)}\|^2.$$

4. if $\frac{D^{(k-1)} - D^{(k)}}{D^{(k-1)}} > \epsilon$, go back to step 1.

5. Set $D^* = D^{(k)}$ and set $q_i^* = q_i^{(k)}$.

Splitting and optimization phases of the algorithm are repeated until the desired number of quantization centers is reached.

REFERENCES

[1] `http://www.data-compression.com/vq.html`.

[2] Y. Linde A., Buzo, and R. Gray. An algorithm for vector quantizer design. *Communications, IEEE Transactions on*, 28(1):84 – 95, jan 1980.

[3] P. Bas, T. Filler, and T. Pevny. "Break Our Steganographic System" — the ins and outs of organizing BOSS. In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Workshop*, Lecture Notes in Computer Science, Prague, Czech Republic, May 18–20, 2011.

[4] S. Boltz, E. Debreuve, and M. Barlaud. High-dimensional statistical distance for region-of-interest tracking: Application to combining a soft geometric constraint with radiometry. In *Proceedings IEEE, Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2007*, pages 1–8, Minneapolis, MN, June 18–23, 2007.

[5] T. Filler, T. Pevný, and P. Bas. BOSS. `http://www.agents.cz/boss/BOSSFinal/`, July 2010.

[6] J. Fridrich. Feature-based steganalysis for JPEG images and its implications for future design of steganographic schemes. In J. Fridrich, editor, *Information Hiding, 6th International Workshop*, volume 3200 of Lecture Notes in Computer Science, pages 67–81, Toronto, Canada, May 23–25, 2004. Springer-Verlag, New York.

[7] J. Fridrich, J. Kodovský, M. Goljan, and V. Holub. Steganalysis of spatially-adaptive steganography. In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Workshop*, Lecture Notes in Computer Science, Prague, Czech Republic, May 18–20, 2011.

[8] G. Gül and F. Kurugollu. A new methodology in steganalysis: Breaking highly undetactable steganograpy (HUGO). In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Workshop*, Lecture Notes in Computer Science, pages 71–84, Prague, Czech Republic, May 18–20, 2011.

[9] A. D. Ker and R. Böhme. Revisiting weighted stego-image steganalysis. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, volume 6819, pages 5 1–5 17, San Jose, CA, January 27–31, 2008.

[10] J. Kodovský and J. Fridrich. Calibration revisited. In J. Dittmann, S. Craver, and J. Fridrich, editors, *Proceedings of the 11th ACM Multimedia & Security Workshop*, pages 63–74, Princeton, NJ, September 7–8, 2009.

[11] J. Kodovský and J. Fridrich. Steganalysis in high dimensions: Fusing classifiers built on random subspaces. In N. D. Memon, E. J. Delp, P. W. Wong, and J. Dittmann, editors, *Proceedings SPIE, Electronic Imaging, Security and Forensics of Multimedia XIII*, volume 7880, San Francisco, CA, January 23–26, 2011.

[12] J. Kodovsky, J. Fridrich, and V. Holub. On dangers of overtraining steganography to incomplete cover model. In *Proceedings of the thirteenth ACM multimedia workshop on Multimedia and security*, MM&Sec '11, pages 69–76, New York, NY, USA, 2011. ACM.

[13] J. KodovskÃœ, J. Fridrich, and V. Holub. Ensemble classifiers for steganalysis of digital media. *IEEE Transactions on Information Forensics and Security*, 2012. To appear.

[14] T. Pevný, P. Bas, and J. Fridrich. Steganalysis by subtractive pixel adjacency matrix. *IEEE Transactions on Information Forensics and Security*, 5(2):215–224, June 2010.

[15] T. Pevný, T. Filler, and P. Bas. Using high-dimensional image models to perform highly undetectable steganography. In P. W. L. Fong, R. Böhme, and Rei Safavi-Naini, editors, *Information Hiding, 12th International Workshop*, Lecture Notes in Computer Science, Calgary, Canada, June 28–30, 2010.

[16] T. Pevný and J. Fridrich. Benchmarking for steganography. In K. Solanki, K. Sullivan, and U. Madhow, editors, *Information Hiding, 10th International Workshop*, volume 5284 of Lecture Notes in Computer Science, pages 251–267, Santa Barbara, CA, June 19–21, 2008. Springer-Verlag, New York.

[17] Y. Q. Shi, C. Chen, and W. Chen. A Markov process based approach to effective attacking JPEG steganography. In J. L. Camenisch, C. S. Collberg, N. F. Johnson, and P. Sallee, editors, *Information Hiding, 8th International Workshop*, volume 4437 of Lecture Notes in Computer Science, pages 249–264, Alexandria, VA, July 10–12, 2006. Springer-Verlag, New York.

[18] H. Singh, N. Misra, V. Hnizdo, A. Fedorowicz, and E. Demchuk. Nearest neighbor estimates of entropy. *American Journal of Math. and Management Sciences*, 23:301–321, 2003.

[19] K. Solanki, A. Sarkar, and B. S. Manjunath. YASS: Yet another steganographic scheme that resists blind steganalysis. In T. Furon, F. Cayre, G. Doërr, and P. Bas, editors, *Information Hiding, 9th International Workshop*, volume 4567 of Lecture Notes in Computer Science, pages 16–31, Saint Malo, France, June 11–13, 2007. Springer-Verlag, New York.

[20] K. Solanki, K. Sullivan, U. Madhow, B. S. Manjunath, and S. Chandrasekaran. Provably secure steganography: Achieving zero K–L divergence using statistical restoration. In *Proceedings IEEE, International Conference on Image Processing, ICIP 2006*, pages 125–128, Atlanta, GA, October 8–11, 2006.

[21] K. Sullivan, U. Madhow, B. S. Manjunath, and S. Chandrasekaran. Steganalysis for Markov cover data with applications to images. *IEEE Transactions on Information Forensics and Security*, 1(2):275–287, June 2006.