Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 855–860
DOI: 10.15439/2015F53
ACSIS, Vol. 5

Using the Generalised Viterbi Algorithm to Achieve a Highly Effective Stegosystem for Images

Valery Korzhik
(Member of IEEE)
State University
of Telecommunications
Saint-Petersburg, Russia
Email: val-korzhik@yandex.ru

Guillermo Morales-Luna
Computer Science
CINVESTAV-IPN
Mexico City, Mexico
Email: gmorales@cs.cinvestav.mx

Ivan Fedyanin
State University of Telecommunications
Saint-Petersburg, Russia
Email: ivan.a.fedyanin@gmail.com

***Abstract*—The HUGO project, published at 2010, can be considered as one of the most promising direction in the design of highly undetectable steganography. The main idea of that approach is to minimise the embedding impact from the steganalysis point of view. This goal is achieved by using trellis codes in the embedding procedure, the Viterbi algorithm (VA) and the SPAM features. But the optimality of VA was kept still unclear because a generic purpose of VA is to correct errors with trellis codes instead of embedding secret information. The first goal of the current paper is to prove the optimality of VA application in its generalised form proposed about 30 years ago by one of the authors of this paper. The second goal is to optimise the parameters of the trellis code check matrix for better undetectability of stegosystems.**

***Index Terms*—stegosystem, images, HUGO project, trellis codes, generalised Viterbi algorithm.**

I. Introduction

STEGANOGRAPHY (SG) is the information hiding technique that embeds the hidden information into an innocent *cover object* (CO) under conditions that CO is not corrupted significantly and that the presence of the additional information in CO may not be detected. This goal entails an obvious requirement: the embedding impact has to be minimised from the steganalysis point of view. Moreover, it does not mean that the number of changes into CO just after embedding should be minimised because the changes "weights" may differ, hence a minimisation of changes into CO does not necessary results in the minimisation of SG detectability.

The assumed most effective stegoanalytic method is the so called *blind steganalysis* based on the transition probabilities of the *Subtractive Pixel Adjacency Matrix* (SPAM) features model between neighboring pixel of the image and CO along 8 different directions [1], [2].

Let $X \in \mathbb{R}^{n_1\times n_2}$ be a grey scale cover image and let $Y \in \mathbb{R}^{n_1\times n_2}$ be the resulting image after embedding using some stego-algorithm. Let $D(X,Y)$ be the distortion of the stegoimage and the cover image, in the sense of ability of the SPAM stegoalgorithm to distinguish X and Y. Up to an enumeration of the pixel array, any image X can be regarded as an array $\mathbf{x} = (x_i)_{i=1}^n$, with $n = n_1 n_2$.

The additive distortion function chosen for SPAM and *Highly Undetectable steGO* (HUGO) [3] is

$$(X,Y) \mapsto D(X,Y) = \sum_{i=1}^{n} \rho_i\,|x_i - y_i| \qquad (1)$$

where, $\rho_i \in \mathbb{R}^+ \cup \{+\infty\}$ is the weight coefficient (the cost of changing the i-th pixel), x_i, y_i are the values of the i-th pixel at X and Y respectively. All changes are restricted to ±1 increments, so that the following inequality holds after ±1-embeddings in LSB:

$$\forall i \text{ with } 1 \le i \le n:\ |x_i - y_i| \le 1.$$

The additive form of (1) means that detectability of SG does not depend on the correlation between the embedded bits. That assumption holds when the changed pixels are located sufficiently far from each other, which in turn holds when the embedding rate is relatively low.

Two immediate problems arise in the design of effective SG:

1) How to choose adequately the weights $(\rho_i)_{i=1}^n$?
2) What is the best coding method for changing pixels according to their weights?

In order to find the pixel weights to be used at (1), for each $i \in \{1,\dots,n\}$, let $Y^{(i)}$ be the image X with the i-th pixel changed. Then, $\forall j \in \{1,\dots,n\}:\ |x_j - y_j^{(i)}| \le 1$. Let us pick $\rho_i = D(X, Y^{(i)})$.

The weight $D(X,Y)$ can be calculated as proposed in [3], as the addition of two sums:

$$\sum_{d_1,d_2=-T}^{T} w(d_1,d_2)\left|\sum_{k\in U}\left(C^k_{d_1d_2}(X) - C^k_{d_1d_2}(Y)\right)\right| + \sum_{d_1,d_2=-T}^{T} w(d_1,d_2)\left|\sum_{k\in V}\left(C^k_{d_1d_2}(X) - C^k_{d_1d_2}(Y)\right)\right| \qquad (2)$$

where the map $C^k_{d_1d_2}$ is calculated, in line with the SPAM features, as the Markov transition probabilities for the eight directions at $U \cup V$, with

$$U = \{\leftarrow, \rightarrow, \uparrow, \downarrow\} \text{ and } V = \{\searrow, \nwarrow, \nearrow, \swarrow\}.$$

In particular, for the case of the horizontal direction ($\rightarrow$), for any integers $d_1, d_2 \in [-T, T]$:

$$C^{\rightarrow}_{d_1 d_2}(X) = \Pr\left(D^{\rightarrow}_{i,j} = d_1 \;\&\; D^{\rightarrow}_{i,j+1} = d_2\right) \quad (3)$$

where for $1 \leq i \leq n_1$ and $1 \leq j \leq n_2 - 1$,

$$D^{\rightarrow}_{i,j} = X_{i,j} - X_{i,j+1}.$$

As in [3], we consider a weight function of the form:

$$w(d_1, d_2) = \left[\sqrt{d_1^2 + d_2^2} + \sigma\right]^{-\gamma} \quad (4)$$

for some optimised parameters $\sigma > 0$ and $\gamma > 0$. It is well known [1] that in order to minimise the number of changes among pixels of the CO, a *syndrome coding* may be used:

$$H\,\mathbf{y} = \mathbf{m} \quad (5)$$

where $\mathbf{y}$ is the block of stego-image bits of length $n = n_1 n_2$, $\mathbf{m}$ is a block of information bits of length k that should be embedded into $\mathbf{y}$ and H is a $(k \times n)$-matrix chosen for encoding. Next, among all the solutions $\mathbf{y}$, for given $\mathbf{m}$ and H, it is necessary to select those providing minimum number of changes between $\mathbf{y}$ and its CO block $\mathbf{x}$. At [1], a technique to solve this problem was presented, and it is especially simple for the Hamming code with a specific check matrix H. But in our case it is necessary to minimise not the number of changes in CO but the distortion function $D : (X, Y) \mapsto D(X, Y)$ given by (1). It seems to be similar to a transition from hard decoding to soft decoding in communications where trellis codes using Viterbi algorithm for decoding can be more favorable than block codes, using some algebraic decoding algorithms. Nevertheless there exists a significant difference in the solution of the matrix equation (5) with respect to $\mathbf{y}$ given $\mathbf{m}$ with minimising the distortion function D given $\mathbf{x}$ as a CO and the solution of the error correction procedure equation $H\,\mathbf{m} = \mathbf{y}$, given some distance between $\mathbf{y}$ and $\mathbf{x}$. Fortunately, there exists the so called *generalised Viterbi algorithm* (GVA) proposed in 1984 [4], [5] that is able to describe both error correction and also steganographic embedding problem in analytic form without execution on trellis graphs.

This method of stegosystem design based on trellis codes and GVA becomes clearer and gets a strict justification. We describe GVA and its application to steganographic problem design in the Section II.

Results of matrix optimisation, which are obtained by simulation with the use of GVA, are presented in Section III. Section IV concludes the paper.

II. Application of GVA to the SG system design problem

Let us state a problem that results in a natural solution within GVA [4], [5].

Problem. Find

$$\tilde{\mathbf{x}}_N = \arg\min_{\mathbf{x}_N} \Lambda_N(\mathbf{x}_N) \quad (6)$$

where

$$\begin{aligned} \Lambda_N(\mathbf{x}_N) &= \sum_{k=1}^{N} \lambda(\xi_k) \quad \text{with} \\ \xi_k &= \begin{cases} (x_1, x_2, \ldots, x_\nu) & \text{if } k \leq \nu \\ (x_{k-\nu}, x_{k-(\nu-1)}, \ldots, x_k) & \text{if } k > \nu \end{cases} \end{aligned}$$

where $0 \leq \nu \leq N-1$, each entry x_j is in the set X, $\mathrm{card}\,(X) = r$, and λ is a real function defined on the set of finite length real sequences. Here, the goal at (6) is to minimise, but it can be to maximise, as well.

The stated problem can be solved through an algorithm introduced at [5]:

1. Find $\tilde{x}_1 = \arg\min_{x_1} \Lambda_{\nu+1}(x_1, x_2, \ldots, x_{\nu+1})$.
2. Find $\tilde{x}_2 = \arg\min_{x_s} \Lambda_{\nu+2}(\tilde{x}_1, x_2, \ldots, x_{\nu+2})$.

$\vdots$

s. Find $\tilde{x}_s = \arg\min_{x_s} \Lambda_{\nu+2}(\tilde{x}_1, \ldots, \tilde{x}_{s-1}, x_s, \ldots, x_{\nu+s})$.

$\vdots$

$N-\nu$. Find $(\tilde{x}_{N-\nu}, \ldots, \tilde{x}_N) = \arg\min_{(x_{N-\nu}, \ldots, x_N)} \Lambda_{\nu+2}(\tilde{\mathbf{x}}_N)$, where $\tilde{\mathbf{x}}_N = (\tilde{x}_1, \ldots, \tilde{x}_{N-\nu-1}, x_{N-\nu}, \ldots, x_{\nu+s})$

It is easy to see that for all steps, except the last one, the number of calculations for every argument is at most $r^{\nu+1}$ and the last step requires at most r^{ν} operations.

We note that the conventional VA is a particular case of GVA with $\nu = 1$. Then we get:

$$\Lambda_N(\mathbf{x}_N) = \lambda(x_1) + \lambda(x_1, x_2) + \cdots + \lambda(x_{N-1}, x_N). \quad (7)$$

If we assume now that each x_k in (7) is the state of trellis on the k-th step and $\lambda(x_k, x_{k+1})$ is the length of branch from the state x_k to the state x_{k+1}, then the decoding problem for convolutional code presented by trellis diagram is equivalent to a minimisation of (6). But fortunately, GVA can be used also for a situation of embedding problem for SG given by (5) that seems to be at a single glance completely different than correction of errors by convolutional codes.

Let us consider a matrix H in (5) that has the "step-wise" sliding form based on a submatrix $\tilde{H}$ of order $t \times w$ pictured at Fig. 1.

In order to simplify further the description, let us consider the particular case of $t = 2$, $w = 2$, and $\tilde{H} = [h_{ij}]_{1 \leq i,j \leq 2}$. Let us assume also $k > 0$, and $n = 2k$. Then equation (5) can be written as follows:

$$\begin{bmatrix} h_{11} & h_{12} & 0 & 0 & \cdots & 0 & 0 \\ h_{21} & h_{22} & h_{11} & h_{12} & \cdots & 0 & 0 \\ 0 & 0 & h_{21} & h_{22} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & h_{11} & h_{12} \end{bmatrix} \mathbf{y} = \mathbf{m} \quad (8)$$

with

$$\begin{aligned} \mathbf{y} &= [y_1 \;\cdots\; y_{2k}]^T \\ \mathbf{m} &= [m_1 \;\cdots\; m_k]^T \end{aligned}$$

$$H = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1w} & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ h_{21} & h_{22} & \cdots & h_{2w} & h_{11} & h_{12} & \cdots & h_{1w} & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & h_{21} & h_{22} & \cdots & h_{2w} & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ h_{t1} & h_{t2} & \cdots & h_{tw} & \vdots & \vdots & & \vdots & & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots & h_{t1} & h_{t2} & \cdots & h_{tw} & & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & & h_{11} & h_{12} & \cdots & h_{1w} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & h_{21} & h_{22} & \cdots & h_{2w} & h_{11} & h_{12} & \cdots & h_{1w} \end{bmatrix}$$

Fig. 1. H as a step-wise sliding matrix.

determining the equation system

$$\begin{aligned} h_{11}y_1 + h_{12}y_2 &= m_1 \\ \forall j = 2, \ldots, k-1: & \\ h_{21}y_{2j-3} + h_{22}y_{2j-2} + h_{11}y_{2j-1} + h_{12}y_{2j} &= m_j \\ h_{11}y_{2k-1} + h_{12}y_{2k} &= m_k \end{aligned} \quad (9)$$

It is possible to apply GVA to solve the system (9) given the column vector M and X providing a minimisation of the weight

$$D(X,Y) = \sum_{j=1}^{2k} \rho_j \, |x_j - y_j|.$$

This algorithm can be performed through the following steps:

1) Build the Table of variants $(\hat{y}_1, \hat{y}_2)$ for all possible tuples (y_3, y_4) satisfying equation (9), for $j = 2$, and minimise the function

$$\Lambda_1 = \rho(x_1, \hat{y}_1) + \rho(x_2, \hat{y}_2),$$

where $\rho(x_i, y_i) = \rho_i |x_i - y_i|$.

2) Build the Table of variants $(\hat{y}_3, \hat{y}_4)$ for all possible tuples (y_5, y_6) satisfying equation (9), for $j = 3$, and minimise the function

$$\Lambda_2 = \rho(x_1, \hat{y}_1) + \rho(x_2, \hat{y}_2) + \rho(x_3, \hat{y}_3) + \rho(x_4, \hat{y}_4),$$

where $\rho(x_i, y_i) = \rho_i |x_i - y_i|$.

Proceed similarly up to the last equation at (9).

Example. Let $k = 3$, $\rho_i = 1$ for $i = 1, \ldots, 6$, $\mathbf{x} = 101110$ and $\mathbf{m} = 100$. Then the matrix H is

$$\begin{aligned} H &= \begin{bmatrix} h_{11} & h_{12} & 0 & 0 & 0 & 0 \\ h_{21} & h_{22} & h_{11} & h_{12} & 0 & 0 \\ 0 & 0 & h_{21} & h_{22} & h_{11} & h_{12} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}. \end{aligned}$$

In line with (8) we have

$$H\,\mathbf{y} = \mathbf{m} \quad (10)$$

where

$$\begin{aligned} \mathbf{y} &= [y_1 \;\; y_2 \;\; y_3 \;\; y_4 \;\; y_5 \;\; y_6]^T \\ \mathbf{m} &= [1 \;\; 0 \;\; 0]^T \end{aligned}$$

y_1	y_2	y_3	y_4	(*)	Λ_1
1	0	0	0	1	–
1	1	0	0	0*	1
1	0	0	1	1	–
1	1	0	1	0*	1
1	0	1	0	0*	0
1	1	1	0	1	–
1	0	1	1	0*	0
1	1	1	1	1	–

(*) Left side of the second equation in (9)
Asterisk means that left side of the second equation of (9) coincides with $m_2 = 0$.

TABLE I
VALUES OF Λ_1.

y_3	y_4	y_5	y_6
0	0	0	0
1	1		
0	0	0	1
1	1		
0	1	1	0
1	0		
0	1	1	1
1	0		

TABLE II
ADMISSIBLE TUPLES (y_1, y_2, y_3, y_4).

1) In order to satisfy the first equation in (9) we get two possibilities $(y_1, y_2) = (1, 0)$, and $(y_1, y_2) = (1, 1)$.
2) For all possible pairs (y_3, y_4) and (y_1, y_2) which are obtained in the Step 1) we get the possibilities to satisfy the second equation in (9) shown at Table I.

In Table I there are presented also the calculation results of the values Λ_1 for "survived" strings (with an asterisk). Next in Table II we present all possible pairs (y_3, y_4) satisfying to the corresponding equation in (9) for every pair (y_5, y_6).

By combining the Tables I and II, we get Table III presenting all possible(y_1, y_2, y_3, y_4) tuples for every pair (y_5, y_6) and corresponding to them, the values Λ_3.

Now it is possible to minimise Λ_3 by selecting a pair (y_5, y_6). This gives two optimal tuples $\mathbf{y} = 101100$ and $\mathbf{y} = 101010$. It is easy to see that each of these tuples requires one change of the tuple $\mathbf{x}$ and satisfies the equation (10).

This approach can be extended to any "step-wise" matrix generated by shifting the $(t \times w)$-submatrix $\tilde{H}$. Then the

y_1	y_2	y_3	y_4	y_5	y_6	Λ_3
1	1	0	0	0	0	4
1	0	1	1			1
1	1	0	0	0	1	5
1	0	1	1			2
1	1	0	1	1	0	2
1	0	1	0			1
1	1	0	1	1	1	3
1	0	1	0			2

TABLE III
VALUES OF Λ_3.

complexity for GVA will be of the order $O(kwt\,2^{wt})$ with respect to operations.

III. OPTIMISATION OF $\tilde{H}$ BY SIMULATION WITH GVA

Let us consider initially the case with $\rho_i = 1$, for $i = 1, \ldots, n$. This means that we try to minimise the number of changes into the image pixels after embedding against the sizes t and w of a randomly chosen submatrix $\tilde{H}$. The results of simulation are shown in Fig. 2 where relative changes

$$\nu = \frac{\text{card}\left(\{i|\ x_i \neq y_j\}\right)}{n}$$

are presented on the vertical axis.

From Fig. 2 it can be seen that it is sufficient to bound the sizes of the submatrix $\tilde{H}$ as $t \leq 15$, $w \leq 10$, at least for the case $\rho_i = 1$, for $i = 1, \ldots, n$.

But it is more interesting to investigate the undetectability of SG based on syndrome embedding under the condition that the SG detection is performed by blind SVM-based steganalysis with the use of SPAM features (see (2)-(4)). The experiment has been arranged as follows: The image base was taken similar to those considered in [6]. Since the embedding procedure is very time-consuming the images were reduced to lesser sizes. The weight coefficients ρ_{ij} in (1) were calculated by (2), (3) with the weight function (4) after optimization of parameters σ and γ. At the training SVM stage both 500 images with and without embedding were used. During the testing SVM stage, there were executed 500 images with and without embedding. This HUGO-based algorithm was compared with conventional ± 1 embedding algorithm taken with the same embedding rate $R = \frac{k}{n}$, where k is the number of embedded bits and n is the number of image pixels. As a criterion of stegosystem undetectability, it was used (in line with a recommendations [1]) the averaged error probability

$$P_e = \frac{1}{2}\left(P_m + P_{fa}\right)$$

where P_m is the probability of SG missing and P_{fa} is the probability of SG false alarm. The value P_e has been minimised at the cost of SVM threshold optimization.

The results of simulations are shown in Tables IV and V.

We can see from these Tables that the use of the HUGO trellis code-based SG offers some advantages in undetectabilities against a conventional ± 1 in LSB embedding SG.

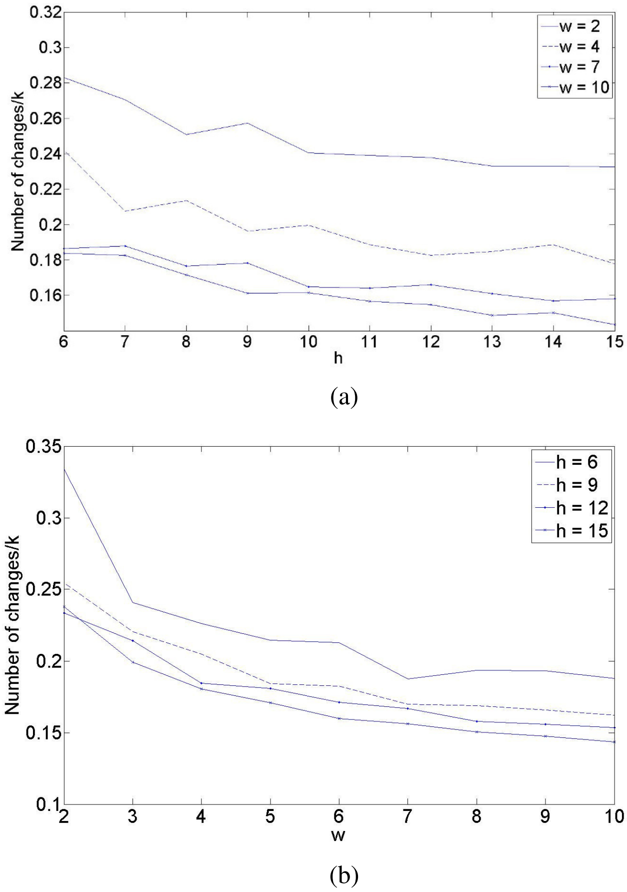

Fig. 2. Relative changes of image pixels after embedding based on trellis codes with $n_1 n_2 = 10^5$. (a) against the matrix parameter t given w, and (b) against the matrix parameter w given t.

Image size	Embedding rate $R = \frac{k}{n}$	P_e
16×16	0.4	0.27
64×64	0.4	0.176
128×128	0.2	0.1780
128×128	0.4	0.123
256×256	0.4	0.099

TABLE IV
THE PROBABILITIES OF INCORRECT ± 1 SG SYSTEM DETECTION (P_e) FOR DIFFERENT IMAGE SIZES AND EMBEDDING RATES R.

It is worth to note that the undetectability of SG, even in the case of a trellis code-based embedding (used with the HUGO project), depends significantly on the image texturing. We have found that the greater is the degree of texture, the most frequent undetectability of the corresponding image. Qualitatively, image texturing means that the image has a presence of precise contours, while not texture images have sliding luminance changing and noisy background. Numerically image texture can be estimated by the parameter [1]

$$t_n = \frac{1}{n_1 n_2} \sum_{ij} \left(\max_k B_{ij}^k - \min_k B_{ij}^k \right) \tag{11}$$

Image size	$R = \frac{k}{n}$	SPAM parameters			P_e
		T	σ	γ	
16×16	0.4	10	10	4	0.337
64×64	0.4	10	10	4	0.24
128×128	0.4	10	10	4	0.162
128×128	0.2	10	10	4	0.269

TABLE V
THE PROBABILITIES OF INCORRECT HUGO SG SYSTEM SPAM-BASED DETECTION (P_e) FOR DIFFERENT IMAGE SIZES AND EMBEDDING RATES R

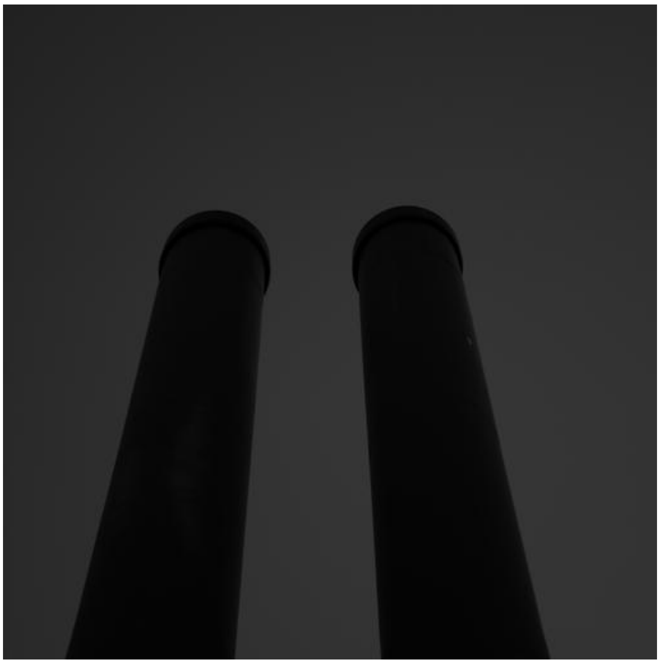

Fig. 3. Example of a low texture image.

Fig. 4. Example of a high texture image.

Image size	t_n	$R = \frac{k}{n}$	SPAM parameters			P_e
			T	σ	γ	
64×64	< 0.194	0.4	3	10	4	0.012
64×64	> 5.28	0.4	3	10	4	0.391

TABLE VI
THE ERROR DETECTING PROBABILITIES FOR HUGO-BASED SG AFTER EMBEDDING OF MESSAGES INTO THE IMAGES WITH DIFFERENT TEXTURING.

where B_{ij}^k is a (2×2)-pixel block with (i, j)-coordinates and k is the k-th pixel of this block.

On Fig's. 3 and 4 there are shown examples of images with low and high texture coefficient t_n, respectively.

As it can be seen by (11), in order to calculate a measure of image texture t_n it is necessary to divide the image on disjoint (2×2)-blocks and next to calculate for every block the difference between maximum and minimum pixel luminance of this block.

In Table VI there are presented the averaged error detecting probabilities for HUGO-based SG after embedding of messages into the images with different texturing. For both SVM training and testing phases 500 images from different image sets were used.

We can see from this Table that, in fact, the level of image texturing affects very significantly on SG system detectability. It seems to be recommended to select for SG embedding such images, which have large texture level. But on the other side it can be looking suspiciously with point of steganographic usage view. Maybe it is better to embed the amount of secret bits depending on the level of image texture.

IV. CONCLUSION

A new generation of stegosystems with the use of trellis code-based embedding is very promising because this approach minimises an embedding impact with the point of view blind SVM-SPAM based steganalysis. This is the so called HUGO project developed recently [3].

But our main contribution into this direction is to make more clear and to prove rigorously that the use of generalised Viterbi Algorithm is in fact the optimal embedding procedure jointly with trellis codes.

With the use of this method we have shown experimentally that HUGO-based SG has significant advantage against simple stegosystems (like LSB or ± 1 algorithm). We have found also that the level of image texturing is very important in a choice of images intended for embedding of hidden messages with high undetectability.

We agree with the importance given to “Open Problem 1” in [7], namely the design of effective coding schemes for non-additive distortion function.

REFERENCES

[1] J. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*, 1st ed. New York, NY, USA: Cambridge University Press, 2009.

[2] T. Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *Information Forensics and Security, IEEE Transactions on*, vol. 5, no. 2, pp. 215–224, June 2010.

[3] T. Pevný, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Proceedings of the 12th International Conference on Information Hiding*, ser. IH'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 161–177.

[4] V. Korzhik, "A generalization of Viterbi algorithm on the case of channel model described by additive Markov channel," in *Proc. of IV Intern'l Symp on Information Theory, Part II*, 1984, pp. 109–111.

[5] V. Korzhik and Y. Lopato, "Optimal decoding of convolutional codes in channels with additive markov noises," in *Proc. of IV Intern'l Symp on Information Theory, Part II*, 1984, pp. 35–40.

[6] P. Bas, T. Filler, and T. Pevný, ""Break our steganographic system": the ins and outs of organizing BOSS," in *Proceedings of the 13th international conference on Information hiding*, ser. IH'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 59–70. [Online]. Available: http://dl.acm.org/citation.cfm?id=2042445.2042452

[7] A. D. Ker, P. Bas, R. Böhme, R. Cogranne, S. Craver, T. Filler, J. Fridrich, and T. Pevný, "Moving steganography and steganalysis from the laboratory into the real world," in *Proceedings of the First ACM Workshop on Information Hiding and Multimedia Security*, ser. IH&MMSec '13. New York, NY, USA: ACM, 2013, pp. 45–58. [Online]. Available: http://doi.acm.org/10.1145/2482513.2482965