SECURE STEGANOGRAPHY: STATISTICAL RESTORATION IN THE TRANSFORM DOMAIN WITH BEST INTEGER PERTURBATIONS TO PIXEL VALUES

A. Sarkar,† *K. Solanki** *and B. S. Manjunath*†

†Department of Electrical and Computer Engineering, University of California at Santa Barbara, Santa Barbara, CA 93106

*Mayachitra Inc. 5266 Hollister Ave. Santa Barbara, CA 93111

ABSTRACT

We consider the problem of *active* steganography, wherein the goal is to survive benign attacks in addition to maintaining statistical and perceptual transparency. The stego image can be "advertised" in uncompressed format, but they can survive JPEG compression upto a certain design quality factor while still resisting statistical steganalysis. The data is hidden in the discrete cosine transform (DCT) coefficients using the *statistical restoration* framework: a fraction of the coefficients are used for hiding and the rest are used for restoring the statistics. In order to advertise the images in uncompressed format, we must restore the *unquantized* or *continuous* statistics. This paper extends the statistical restoration framework so as to make only integer perturbations to the pixel values while modifying the transform coefficients, thus matching their histogram computed using very small bin-width. We present numerical results confirming the applicability of the presented technique.

Index Terms— Steganography, steganalysis, fractional bin-width, optimal pixel perturbations, statistical restoration

1. INTRODUCTION

We consider the problem of *active* steganography in this paper, wherein the goal is to transmit a *message* by secretly embedding it into a *cover* (the host signal) in such a way that (i) the very existence of the hidden message is not revealed to a third party and (ii) the message can be recovered even after benign or malicious processing of the *stego* signal (host with embedded data). Although most prior works have focussed on *passive* steganography in which no attacks are assumed (see, for example, [1, 2, 3]), we believe that robustness against "benign" processing such as compression or additive noise is highly desirable. For instance, a digital image with hidden content may get compressed as it changes hands, or if it goes over a low bandwidth link such as a wireless channel. It is to be noted that though we do not explicitly model attacks in this paper, the presented method is designed to survive compression and other mild distortion-constrained attacks without giving up its steganographic security, and hence it fits into the active steganography model.

Specifically, we look at the problem of surviving JPEG compression while "advertising" the image in uncompressed format and maintaining *statistical* as well as *perceptual* transparency. To achieve this, we follow the framework of *statistical restoration* proposed in our prior work [4, 5], wherein a fraction of the host symbols are used for hiding and the rest are used to *restore* the statistics of the stego to match that of the original host.

This research is supported in part by a grant from ONR # N00014-05-1-0816.

In order to advertise the images in uncompressed format, we must restore the *unquantized* or *continuous* statistics in the transform domain. Such a scheme employing statistical restoration with quantization index modulation (QIM) hiding was first demonstrated in [4]. The results showed much improvement over standard QIM hiding, but the steganography was still detectable more than 20% of the time. The reason is the presence of unavoidable **round-off errors** since the pixel values must always be integers. *For restoring the continuous statistics, we must consider histograms computed with very small bin-widths, which makes the system susceptible to round-off errors*. This is a practical limitation that must be considered by any steganographic scheme embedding data in the transform domain. There is always a risk of leaking the presence of embedded data due to the fact that these round-off errors modify the statistics of the transform coefficients.

In this paper, we present a method that allows us to make integer modifications to the pixel values during the statistical restoration of the transform coefficients. The price we pay is the lowered rate of embedding. Though the general ideas can be applied to restoration in any transform domain or with any hiding method (QIM or spread spectrum), we present in this paper the specific case of QIM hiding in the discrete cosine transform (DCT) domain, thus surviving JPEG compression upto a certain quality factor (QF) by design.

A limitation of the presented method is the possibility of it being detected by recent steganalysis techniques ([6, 7]) that exploit cover memory. While our recent work on second-order statistical restoration addresses these issues [8], the main contribution of the present paper is a method allowing integer perturbations to the pixels while embedding and restoring transform coefficient histograms with very fine bin-widths. Extending the presented scheme for higher order steganalysis would be an interesting avenue for future work.

Another impending issue is the choice of bin-width. Scott [9] proposes a framework for computing the optimal histogram bin-width for data analysis. For the DCT coefficients, Scott's framework recommends bin-width values of the order of 10^{-4} (not considering noise due to round-off). Using our method, we could restore the DCT histogram upto a bin-width of 10^{-2}, which makes the scheme vulnerable to steganalysis. In spite of the limitations mentioned in this and the previous paragraph, we believe that the presented approach provides a fair contribution, as it solves an important practical problem not previously considered in the literature.

2. PROBLEM FORMULATION

Let us define three discrete cosine transform (DCT) based features - "actual", "quantized" and "unquantized" DCT. The DCT terms computed per 8×8 block are the "actual" DCT terms. After dividing the actual DCT terms by a certain quality factor matrix (without rounding the resultant terms), we obtain the "unquantized" DCT (UQDCT) terms. On rounding them, we obtain the "quantized" DCT (QDCT) terms.

In the statistical restoration framework, certain 8×8 blocks are used for hiding and the rest for compensation. Let X denote the matrix of image pixels, *belonging to compensation blocks*, and ψ denote the 2D-DCT transform, performed per 8×8 block in X, which converts X to Y, as in (1). The matrix of DCT terms Y is then transformed to the UQDCT matrix Z - every 8×8 block in Y, denoted by $Y^{8\times 8}$, is divided element-wise by the 8×8 JPEG quantization matrix M, corresponding to a certain quality factor QF, to produce $Z^{8\times 8}$, the corresponding 8×8 block in Z, as shown in (1).

$$Y = \psi(X),\ Z_{ij}^{8\times 8} = \frac{Y_{ij}^{8\times 8}}{M_{ij}},\ 1 \le i, j \le 8 \tag{1}$$

where M_{ij} is the element in the i^{th} row and j^{th} column of M.

Let us now outline the histogram matching problem in the statistical restoration framework. We use a pre-decided group of elements per 8×8 UQDCT block for hiding and compensation and call this feature set D. Let D be divided into two disjoint sets - H and C, which are meant for hiding and compensation, respectively, as shown in (2), and are generated from different blocks. The compensation set C is thus a subset of the UQDCT set Z. We divide the set D into 1-D bins and find their respective bin-counts (number of terms per bin). We use $B_D(i)$ to denote the bin-count of the i^{th} bin of D. Since the normalized bin-count gives the probability mass function (PMF), compensating for the bin-counts is equivalent to restoring the PMF. Data embedding is performed using dithered QIM in the hiding blocks to change H to H'. For maintaining the perceptual transparency, we only embed data in UQDCT terms of magnitude greater than 0.5, the motivation being the Selective Embedding in Coefficients (SEC) scheme proposed in [10]. The goal of statistical restoration is to change the compensation terms C to C' such that $P_{D'}$, the PMF of D', obtained from D after hiding and compensation, is same as that of D.

$$D = H \cup C,\ D' = H' \cup C' \tag{2}$$

$$H \cap C = \phi \Rightarrow B_D(i) = B_H(i) + B_C(i),\ \forall\, i \tag{3}$$

$$H' \cap C' = \phi \Rightarrow B_{D'}(i) = B_{H'}(i) + B_{C'}(i),\ \forall\, i \tag{4}$$

$$\text{To obtain } P_D = P_{D'},\ \text{we need } B_{D'}(i) = B_D(i),\ \forall\, i \tag{5}$$

$$\Rightarrow B_{C'}(i) = B_C(i) + B_H(i) - B_{H'}(i),\ \forall\, i \tag{6}$$

$B_{H'}$ is computed from the UQDCT stream obtained from the hiding blocks in the stego image and not from the hidden stream in the actual image - this is done to avoid errors in UQDCT computation (causing errors in computation of $B_{H'}$) because of pixel-rounding in the hiding blocks. If $B_{C'}(i)$, the desired bin-count for the i^{th} bin of the compensation coefficients, is negative, compensation is not possible for that bin.

For histogram computation, we consider only those terms that lie in a desired frequency band and have magnitude less than or equal to a certain threshold T. Since the distribution of the UQDCT coefficients peaks sharply near 0 and falls off sharply for higher values, higher valued terms may be ignored in the histogram computation.

Let the new set of UQDCT terms for the compensation blocks be Z' (the set of the desired compensation terms for these blocks is C'). In (7), the change in pixel values of X required to exactly convert Z to Z' is ΔX. The matrix of DCT terms Y in (1) gets changed to Y'.

$$Y' = \psi(X + \Delta X),\ Z_{ij}'^{8\times 8} = \frac{Y_{ij}'^{8\times 8}}{M_{ij}},\ 1 \le i, j \le 8 \tag{7}$$

Now, the exact value of the pixel changes ΔX may also involve some fractional terms. Once we round off the pixel changes, the resulting Z' may no longer satisfy the histogram matching constraint. *The aim is to find a set of integer ΔX values which ensures the histogram matching in UQDCT domain*. A smaller perturbation in Z due to rounding of the pixel changes may offset the histogram matching for smaller bin-widths.

3. ESTIMATION OF THE BEST PIXEL PERTURBATIONS

In (1), X represents the entire pixel matrix *for the compensation blocks*. For simplicity, let us consider one 8×8 block of the image and treat it as X. Similarly, Z, in (1), can be considered as the corresponding 8×8 UQDCT matrix. Though X and Z are 8×8 matrices, we can also represent them by 64-dimensional vectors $\vec{X}$ and $\vec{Z}$, as in (8). While converting the 8×8 matrix to a 64-dimensional vector, we read the matrix column-wise. The conversion from X to Z in (1) can be represented by a matrix A (2D-DCT followed by element-wise division by the JPEG quantization matrix M). In (8), A is a 64×64 matrix while $\vec{X}$ and $\vec{Z}$ are 64×1 vectors.

$$A\vec{X} = \vec{Z} \tag{8}$$

We compute A by using the superposition principle. Say, we set the i^{th} term in the column vector $\vec{X}$ to 1, while the rest are 0, and perform 2D-DCT followed by component-wise division by M to obtain $\vec{Z}$. Thus, $A_{ji} = \vec{Z}_j, 1 \le j \le 64$, where A_{ji} is the element in the j^{th} row and i^{th} column of A and $\vec{Z}_j$ is the j^{th} element of $\vec{Z}$. We vary i 64 times to compute all the elements in A. The matrix A is invertible since both the operations which constitute A, 2D-DCT and component-wise division by M (no rounding off), are reversible.

Out of the 64 terms per compensation block, we compensate using a certain mid-band of UQDCT coefficients. Z consists of two disjoint sets - Z_c and Z_{nc}. Z_c refers to the compensation terms of Z lying in this select band of N_c coefficients, which are considered for histogram computation, while Z_{nc} refers to the rest of the terms in the block (non-compensation terms) - N_c equals 19 in Fig. 1 where Z_c and Z_{nc} are demarcated.

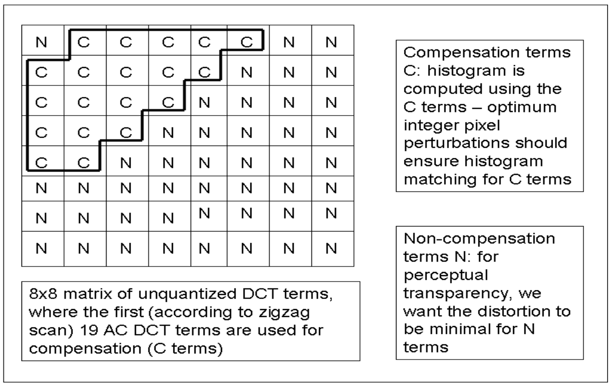

Fig. 1. Compensation (Z_c terms denoted by C) and non-compensation terms (Z_{nc} terms denoted by N) per 8×8 DCT block

Let the vectors $\Delta\vec{Z}_1$ and $\Delta\vec{Z}_2$ denote the limits for the perturbation of the terms in $\vec{Z}$. Perturbation of the elements of $\vec{Z}$ and $\vec{X}$ are denoted by $\Delta\vec{Z}$ and $\Delta\vec{X}$, respectively. For notational convenience, we use $\vec{\le}$ to denote term-by-term inequality between 2 vectors.

$$\vec{F}\ \vec{\le}\ \vec{E} \Rightarrow \vec{F}_i \le \vec{E}_i, 1 \le i \le N \tag{9}$$

where vectors $\vec{F}$ and $\vec{E}$ are each of length N and $\vec{F}_i$ is the i^{th} element of $\vec{F}$. Since A is invertible, the pixel perturbation limits $\Delta\vec{X}_1$ and $\Delta\vec{X}_2$ can be obtained as follows:

$$\Delta\vec{Z}_1 \lesseqqgtr \Delta\vec{Z} \lesseqqgtr \Delta\vec{Z}_2 \Rightarrow$$

$$\Delta\vec{Z} = A\Delta\vec{X} = \alpha(\Delta\vec{Z}_1) + (1-\alpha)(\Delta\vec{Z}_2), \alpha \in [0,1] \quad (10)$$

$$\Rightarrow \Delta\vec{X} = \alpha(A^{-1}\Delta\vec{Z}_1) + (1-\alpha)(A^{-1}\Delta\vec{Z}_2), \alpha \in [0,1] \quad (11)$$

$$\text{Bounds on } \Delta\vec{X}\text{: } \Delta\vec{X}_1 = A^{-1}(\Delta\vec{Z}_1),\ \Delta\vec{X}_2 = A^{-1}(\Delta\vec{Z}_2) \quad (12)$$

(10) follows from the property of convex sets. Since pixel changes must be integers, $\Delta\vec{X}$ should be a rounded version of a weighted average of $\Delta\vec{X}_1$ and $\Delta\vec{X}_2$, i.e.

$$\Delta\vec{X} = \text{round}(\alpha\Delta\vec{X}_1 + (1-\alpha)\Delta\vec{X}_2),\ 0 \le \alpha \le 1, \quad (13)$$

Considering ΔZ_1 and ΔZ_2 as 8×8 matrices, we now explain how their compensation ($\Delta Z_{i,c},\ i = 1,2$) and non-compensation terms ($\Delta Z_{i,nc},\ i = 1,2$) are determined. For every compensation term $z \in Z_c$, we have a corresponding $c' \in C'$, where C' denotes the set of the desired compensation terms, as in (2); let $c' \in [iW, (i+1)W]$, i.e. c' lies in the i^{th} bin where the bin-width is W. The limits of the allowed perturbation Δz for z are then $[iW - z, (i+1)W - z]$. Let $\Delta z_1 = (iW - z)$ and $\Delta z_2 = ((i+1)W - z)$ be the corresponding elements in $\Delta Z_{1,c}$ and $\Delta Z_{2,c}$, respectively. This ensures that $\Delta\vec{Z}_{1,c} \lesseqqgtr \Delta\vec{Z}_c \lesseqqgtr \Delta\vec{Z}_{2,c}$ holds true.

For the Z_{nc} terms, we have set a perturbation range of $[-\delta, \delta]$ (in our experiments, we have used δ = 0.1) - the goal being to minimize the perturbation to maintain good perceptual quality. The perturbation limit for a 8×8 matrix is obtained by combining the limits for the compensation and non-compensation terms: $\Delta Z_i = \Delta Z_{i,c} \cup \Delta Z_{i,nc},\ i = 1,2$. There are two requirements from the UQDCT ΔZ terms in the compensation blocks - histogram matching for the compensation terms (ensured by $\Delta\vec{Z}_{1,c} \lesseqqgtr \Delta\vec{Z}_c \lesseqqgtr \Delta\vec{Z}_{2,c}$) and minimum perturbation, obtained by absolute summation over the perturbations, for the non-compensation terms, as in (14).

We now describe how we meet these requirements. Given a 8×8 matrix X, we can determine $\Delta\vec{Z}_1$ and $\Delta\vec{Z}_2$ (the sequence being: $X \to Z \to Z' \to \Delta Z_1$ & ΔZ_2), and from them $\Delta\vec{X}_1$ and $\Delta\vec{X}_2$, using (12). For finding the best pixel perturbation, we need the optimal value of the weight α in (13). For a certain weighting factor α, we find $\Delta\vec{X}$ using (13). From $\Delta\vec{X}$, we compute $\Delta\vec{Z}$ (=$A\Delta\vec{X}$) and subsequently, $\Delta\vec{Z}_c$ and $\Delta\vec{Z}_{nc}$. We vary α from 0 to 1 in steps of 0.01 and using (14), we choose the optimal weight α_{opt}. Thus, the best pixel perturbation $\Delta\vec{X}_{opt}$ is $\Delta\vec{X}$ computed using $\alpha = \alpha_{opt}$.

$$\alpha_{opt} = \arg\min_{\alpha}\{\sum_{z \in \Delta Z_{nc}} |z|\} \text{ under the constraint that}$$
$$\{\Delta\vec{Z}_{1,c} \lesseqqgtr \Delta\vec{Z}_c \lesseqqgtr \Delta\vec{Z}_{2,c}\} \text{ holds true} \quad (14)$$

If only the histogram matching condition is satisfied, after varying α over [0,1], we choose the α that provides the minimum perturbation, even though it may exceed δ=0.1 for individual ΔZ_{nc} terms. If we fail to obtain perfect histogram matching, for any $\alpha \in$ [0,1], we compute ΔX using that α which provides the minimum perturbation and also reduces the number of histogram mismatches.

4. EXPERIMENTS AND RESULTS

For the steganalysis experiments, we use a total of 4500 images. We use half of them for training and the other half for testing. Both the training and testing sets have half the images as cover and the other half as stego. For testing the accuracy of our fractional bin-width based steganographic methods, we use support vector machine (SVM) based steganalysis. During the training phase, we develop separate SVM classifiers trained on each feature set used for steganalysis. The SVM classifiers are then used to distinguish between cover and stego images in the testing phase.

We use 15% of the 8×8 blocks for hiding and the rest for compensation. From each block, the first 19 AC DCT coefficients (with values in the range $[-T, T]$ where T is the threshold), that occur during a zigzag scan, as shown by the C terms in Fig. 1, are used for histogram computation. The steganographer may use different quality factor matrices to generate the UQDCT terms, different thresholds (T) and different bin-widths (W) for the UQDCT histogram computation. The steganalyst can use his own choice of these parameters. Therefore, for a given set of steganographic parameters, we experiment with a range of steganalytic parameters (Tables 1-4). Also, we study steganalysis results obtained using the histogram of the actual DCT coefficients as features (Table 2). Intuitively, the detectability for histogram-based steganalysis should be higher in the actual DCT domain, as compared to UQDCT, due to the higher dynamic range.

In Tables 1-4, P_{FA} and P_{miss} denote the probability of false alarm and of missed detection, respectively. P_{total} gives the total detection error ($P_{FA} + P_{miss}$). For undetectable hiding, the detector is reduced to random guessing and P_{total} will be close to 1. The design parameters - quality factor, threshold and bin-width used for steganography are denoted by QF_d, T_d and W_d, respectively. For steganalysis, the corresponding terms are QF_s, T_s and W_s, respectively. The steganalyst may JPEG-compress the images (we assume the JPEG attack quality factor $QF_a = QF_s$) before statistical analysis or he may directly analyze the uncompressed images provided by the steganographer - the detection results for these cases are denoted by "JPEG" and "Noise Free", respectively, in Tables 1-4. Except in Table 4, T_s is not explicitly mentioned - ideally, the steganalyst can consider an arbitrarily high threshold for better detection. However, given the size of the training set (2250 images), we limit the feature vector size (equal to the number of histogram bins = $2\lceil\frac{T_s}{W_s}\rceil$) to 400 for proper training of SVMs. Thus, given a bin-width W_s, the maximum allowed value of T_s can be found out.

We experimentally show in Table 1 that the integer pixel perturbation based scheme proposed in this paper improves upon the statistical restoration based scheme without integer perturbations [4, 5] when the fractional bin-width based UQDCT histogram is used for steganalysis. Though the high resolution histogram was compensated for during the statistical restoration process, the **round-off errors** lead to histogram mismatches in the UQDCT domain. The integer perturbation scheme counters the round-off effects and hiding is almost undetectable, as shown by the much higher value of P_{total}.

We also test for recoverability of the embedded data after JPEG-compressing the image at a quality factor QF_a, which is greater than or equal to the design quality factor QF_d. Since the SEC [10] based hiding scheme is adaptive, we use a error-correcting code (ECC) based framework for data retrieval. For QF_d and QF_a both equal to 50, we can embed 7000 bits on an average (using 15% hiding) for 512×512 images, the results being averaged over 100 images. The average bit error rate (BER) is 10^{-2}, without using the ECC framework. For QF_d=50 and QF_a=75, the BER (without ECC) decreases to 10^{-3}. For a non-adaptive scheme, we can use the range [-0.5,0.5] for data hiding: we achieve a higher embedding rate at the cost of increased perceptual distortions.

Table 1. Comparing the performance of the steganography scheme (denoted by "Old") used in [4, 5] with the pixel perturbation scheme (denoted by "New") proposed here, for fractional bin-width based steganalysis in UQDCT domain - for steganography, we use W_d=0.10, T_d=30 and QF_d=50. QF_s is set to 50.

Steganalysis	Noise Free (Old)			Noise Free (New)		
W_s	P_{FA}	P_{miss}	P_{total}	P_{FA}	P_{miss}	P_{total}
1.00	0.21	0.15	0.36	0.48	0.52	1.00
0.50	0.15	0.17	0.32	0.54	0.44	0.98
0.10	0.25	0.12	0.37	0.93	0.07	1.00

Table 2. Actual DCT histogram based steganalysis - we use T_d=30, QF_d=50 and vary W_d. The feature used for steganalysis is the DCT histogram computed using a bin-width of 0.5 considering values in the range [-200,200]. JPEG compression occurs at QF=50.

Steganography	Noise Free			JPEG		
W_d	P_{FA}	P_{miss}	P_{total}	P_{FA}	P_{miss}	P_{total}
0.50	0.00	0.00	0.00	0.01	0.05	0.06
0.10	0.00	0.01	0.01	0.50	0.22	0.72
0.02	0.36	0.53	0.89	0.19	0.65	0.84

Table 3. Variation of detectability with bin-width, using the UQDCT histogram for steganalysis. For steganography, we use T_d=30 and QF_d=50. QF_s is set to 50. For different W_d used for steganography, we vary the W_s used for steganalysis.

Steganography	Steganalysis	Noise Free			JPEG		
W_d	W_s	P_{FA}	P_{miss}	P_{total}	P_{FA}	P_{miss}	P_{total}
0.10	0.10	0.93	0.07	1.00	0.58	0.26	0.84
0.10	0.05	0.00	0.02	0.02	0.53	0.32	0.85
0.50	0.50	0.84	0.15	0.99	0.11	0.09	0.20
0.50	0.25	0.00	0.06	0.06	0.08	0.14	0.22

Table 4. Variation of detectability with threshold, using the UQDCT histogram for steganalysis. For steganography, we use QF_d=75, T_d=30 and W_d=0.10. For steganalysis, we use the same bin-width and quality factor and vary T_s.

Steganalysis	Noise Free			JPEG		
T_s	P_{FA}	P_{miss}	P_{total}	P_{FA}	P_{miss}	P_{total}
20	0.68	0.26	0.94	0.03	0.79	0.82
25	0.73	0.18	0.91	0.02	0.79	0.81
30	0.34	0.21	0.55	0.08	0.50	0.58

4.1. Discussion

For actual DCT histogram based steganalysis, the matching of the DCT histogram requires a high resolution matching in UQDCT domain - therefore, detection for uncompressed and JPEG images is avoided for a bin-width of 0.02 (Table 2). However, when the steganographer uses slightly larger bin-widths ($\approx$ 0.1), the hiding becomes nearly undetectable only after JPEG compression. When a higher bin-width ($\approx$ 0.5) is used during steganography for UQDCT domain steganalysis, the reverse happens - hiding is not detected in the distortion-free case but gets detected after JPEG attacks (Table 3). For uncompressed images, UQDCT domain steganalysis succeeds if the bin-width is lower than that used for steganography (Table 3). For JPEG images, detection is avoided even when a lower bin-width is used for steganalysis than for steganography, i.e. $W_s < W_d$ (Table 3). Also, the detection rate increases (P_{total} decreases) when the steganalyst uses a higher threshold (Table 4) - a larger threshold implies a more detailed histogram which can better capture mismatches between original and stego image histograms. Thus, from a steganalyst's perspective, he should always use a very small bin-width and a high threshold for histogram computation. However, the steganalyst cannot make the bin-width arbitrarily low or the threshold arbitrarily high because that makes the feature vector dimension too high for proper training of the SVMs, unless the size of the training image set is proportionately large.

5. CONCLUSIONS

We have demonstrated a practical method allowing active image steganography that ensures steganographic security while being robust to compression attacks. The images with hidden content can be advertised in uncompressed format without being detected. This has been made possible by an approach that allows integer perturbations to pixel values while matching the transform coefficient histograms at very fine resolutions. Avenues of future work include improving the hiding capacity and incorporating the presented approach into stego schemes that preserve second-order statistics [8].

References

[1] J. Fridrich, M. Goljan, P. Lisoněk, and D. Soukal, "Writing on wet paper," in *ACM Workshop on Multimedia and security*, Magdeburg, Germany, Sept. 2004.

[2] Phil Sallee, "Model-based methods for steganography and steganalysis," *International Journal of Image Graphics*, vol. 5, no. 1, pp. 167–190, 2005.

[3] P. Guillon, T. Furon, and P. Duhamel, "Applied public-key steganography," in *Proc. of SPIE*, San Jose, CA, 2002.

[4] K. Solanki, K. Sullivan, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Statistical restoration for robust and secure steganography," in *Proc. ICIP*, 2005, pp. II–1118–21.

[5] K. Solanki, K. Sullivan, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Probably secure steganography: Achieving zero K-L divergence using statistical restoration," in *Proc. ICIP*, 2006, pp. 125–128.

[6] T. Pevny and J. Fridrich, "Merging Markov and DCT features for multi-class JPEG steganalysis," in *Proc. of SPIE*, San Jose, CA, 2007.

[7] Y. Wang and P. Moulin, "Steganalysis of block-DCT image steganography," in *IEEE workshop on Statistical Signal Processing*, St Louis, MO, USA, Sept. 2003.

[8] A. Sarkar, K. Solanki, U. Madhow, S. Chandrasekaran, and B. S. Manjunath, "Secure steganography: Statistical restoration of the second order dependencies for improved security," *accepted for publication in ICASSP*, 2007.

[9] D. W. Scott, "On optimal and data-based histograms," in *Biometrika*, 1979, vol. 66, pp. 605–610.

[10] K. Solanki, N. Jacobsen, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Robust image-adaptive data hiding based on erasure and error correction," *IEEE Trans. on Image Processing*, vol. 13, no. 12, pp. 1627 –1639, Dec 2004.