BLIND COMPRESSED IMAGE WATERMARKING FOR NOISY COMMUNICATION CHANNELS

Jonathan Mei

Carnegie Mellon University
Department of Electrical & Computer Engineering
Pittsburgh, PA 15213
jmei@cmu.edu

Scott Pudlewski

Lincoln Laboratory
Massachusetts Institute of Technology
Lexington, MA 02420
scott.pudlewski@ll.mit.edu

ABSTRACT

Visually-distorted images can contain valuable information. Indeed, in tactical MANET networks where throughput is extremely valuable and difficult to come by, guaranteeing the delivery of every packet of an encoded image is impractical. However, designing a watermark that is resilient to the types of visual distortion imparted on an image or video due to channel losses is a difficult task. In this work, we introduce a new watermarking scheme for JPEG-compressed images that incorporates ideas from compressed sensing (CS) to achieve robustness against certain types of errors induced by noisy communication channels.

This work uses CS techniques to embed a sparse watermark into L randomly selected quantized JPEG image coefficients. Sparse reconstruction techniques are then used to reconstruct the watermark from the coefficients that were received, including those that were *incorrectly* decoded. Through the development of this watermarking scheme, we would like to demonstrate and explore the effect of the error resilience properties of CS encoded signals on the image watermarking problem.We show through simulation that even with significant visual distortion in the received image, the CS encoded watermark can be detected with very high probability.

Index Terms— Watermarking, Error Coding, Compressed Sensing, JPEG Compression, Erasure Channel

1. INTRODUCTION

Robust digital image watermarking [1], or the act of hiding a hidden signal within an encoded image that is resilient to both noise and tampering, has applications in both commercial and military systems. In this work, we will focus on military applications where an image is generated at some remote imaging sensor (i.e., a persistent surveillance system or an unmanned aerial vehicle (UAV)) and transmitted in part over very lossy tactical MANET links. While traditional digital signatures can be used to verify that an image was *received from* the stated sender, this may not go far enough to ensure that the image was *created by* the sender. In addition, it may be useful to be able to verify the source of an image long after that images was transmitted, for instance in legal proceedings or to determine the source of an information leak. Based on these observations, the watermark should be embedded within the image data so that after any transformation or distortion that produces a *useful image*, the watermark is preserved.

In this work, we explore the use of compressed sensing (CS) [2, 3, 4] in robust watermarking of JPEG encoded images [5]. While there are a number of alternatives that have shown promise in these scenarios (specifically spread spectrum techniques [6, 7]), we believe that the combination of the error resiliency of CS encoded signals along with the minimal complexity required for CS encoding [8] justify its consideration as a watermarking technology.

Compressed sensing is a compression scheme in which sparse signal reconstruction techniques are used to reconstruct a sparse message signal $\mathbf{x}$ from a small number of noise-like random linear combinations $\mathbf{y} = \Phi\mathbf{x}$, where $\Phi \in \mathbb{R}^{M\times N}$, $M < N$ is a noise-like sampling matrix. While CS as a compression technique has a number of obvious applications in multimedia signal compression [9], in this work we are going to explore the impact of two other properties of CS encoded signals. Specifically, we will look at the *error resilience* of CS encoded signals, along with the ability to *interchange received CS samples*, which removes the dependence on any one specific sample. In this work, we will show that these two properties can be used to design a watermarking system that is surprisingly resilient to packet erasures. The goal of this work is to determine whether the error resilience properties of a CS encoded watermark justify the design of yet another data hiding scheme. We believe that, based on the resilience to extremely high error rates within the JEPG encoded image, future work is indeed justified.

Distribution A:Public Release. This work is sponsored by the Department of Defense under Air Force Contract FA8721-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the author and are not necessarily endorsed by the United States Government.

The remainder of this paper is organized as follows. In Section 2, we explain the structure of the watermark and the design of watermarking parameters, while Section 3 explains the detection scheme. Section 4 describes the experimental setup and discuss the simulation results. Finally, we discuss future work and draw the main conclusions in Section 5.

2. WATERMARK DESIGN

2.1. Encoding the Watermark

To create the watermark, we generate a message vector $\mathbf{x} \in \{-1, 0, 1\}^N$ such that $\|\mathbf{x}\|_0 = K$, $N \gg K$. Let $A \in \mathbb{R}^{L \times N}$ be a full rank matrix with $L > N$. Such a matrix can be created with full rank with high probability [3] using a seeded RNG to generate each entry. The watermark vector $\mathbf{y}$ is then generated as $\mathbf{y} = A\mathbf{x}$.

By design of the A matrix, $\mathbf{y}$ is a real-valued vector, while the image coefficients $\mathbf{z}$ are quantized and coded digitally. Thus, in order to practically implement the watermarking scheme, we need to quantize $\mathbf{y}$. Using a pre-computed codebook with Q_n quantization steps, we can find the quantized watermark vector $\mathbf{y}^{(q)} = \mathbf{y} + \eta$, such that $0 \leq Q(y_i^{(q)}) < Q_n$ for $i \in \{0, 1, \ldots, L-1\}$, where $Q(\cdot)$ is the quantization function and η represents the noise introduced by the quantization step.

There are various options to create the watermarked coefficients $w(\cdot)$. Our implementation uses integer modulo masking:

$$w_i(z) = \left\lfloor \frac{z}{Q_n} \right\rfloor + Q(y_i^{(q)}) \tag{1}$$

for $i \in \{0, 1, \ldots, L-1\}$. This is analogous to k least significant bit masking when $Q_n = 2^k$ for integer k.

2.2. Embedding the Watermark

In this work we embed the watermark vector into a JPEG encoded image. However, the basic watermarking framework is not specific to the compression technique. The watermark can be embedded into any set of image transform coefficients (or even the original image). We chose to use the JPEG standard primarily for its popularity.

In JPEG encoding, the DCT coefficients within each of the J macroblocks are scaled through element-wise integer division with quality matrix $B(q)$, where $0 < q \leq 100$ is the quality level of the image. To embed the watermark, we modify these scaled block-DCT coefficients of macroblock intensities. In particular, we modify the subset $\mathcal{S}$ of the J scaled DC coefficients $\mathbf{z}$ of this block-DCT (the DC coefficient of each macroblock is the lowest frequency component of the DCT of that macroblock).

Let $\mathcal{S} \subset \{0, 1, \ldots, J-1\}$ such that $|\mathcal{S}| = L$, and $\mathcal{S}(i)$ denotes the ith element of $\mathcal{S}$. Set $\mathcal{S}$ of L image coefficient indices can be randomly selected using the seeded RNG. The vector $\tilde{\mathbf{z}}$ represents the quantized DC coefficients with the watermark will be embedded. The coefficients $\tilde{z}_{\mathcal{S}(i)} = w_i(z_{\mathcal{S}(i)})$ for $i \in \{0, 1, \ldots, L-1\}$ are watermarked, while the coefficients $\tilde{z}_j = z_j$ for $j \notin \mathcal{S}$ remain unmodified. By watermarking a subset of all coefficients, our scheme allows for some additional security against falsifying. However, note that if $L = J$, then this is equivalent to modifying all coefficients.

Once the watermarked quantized coefficients $\tilde{\mathbf{z}}$ have been computed, they are compressed using variable-length Huffman encoding and periodically marked with reset markers (RST) (to mark the points at which the Huffman decoding process can realign in the case of corrupted or missing bits).

3. WATERMARK DETECTION

In this work we assume the worst-case scenario in that our watermark decoder has no cross-layer information to determine which packets were discarded at the lower layers. While some of the corrupted macroblocks will have a zero amplitude and can therefore easily be removed, we can clearly see in Fig. 1 that many packet losses will result in a "valid" but incorrect decoded macroblock. While it would be feasible to design a parity based detection scheme to find the errors in CS encoded signal (as in [10]), we show that even in this worst-case scenario our scheme is able to maintain high detection rates at high packet error rates with moderate sparsity and expansion levels.

After having removed the detected corrupt macroblocks, we have M watermarked macroblocks remaining. Due to the use of RST, the decoder can determine which M of the remaining blocks originally contained the embedded watermark coefficients. As long as enough blocks are received such that $M \gtrsim K \log(N)$, the receiver can correctly reconstruct the watermark.

3.1. Decoding the Watermark

Let $\mathcal{U}$ be the set of indices of the M watermarked block-DCT coefficients not classified as corrupted after detection, and $\mathcal{U}(i)$ be the i^{th} element of $\mathcal{U}$. The received signal $\hat{\mathbf{y}}$ is then defined element-wise as

$$\hat{y}_i = Q^{-1}\left(\hat{z}_{\mathcal{U}(i)} - \left\lfloor \frac{\hat{z}_{\mathcal{U}(i)}}{Q_n} \right\rfloor\right). \tag{2}$$

Then let $\hat{\mathbf{A}} = \mathbf{A}(\mathcal{U}, :)$ be the corresponding matrix with row indices $\mathcal{U}$ from $\mathbf{A}$. In other words, the missing samples (those that were detected as incorrect) are removed from the set of received samples *along with the corresponding rows in the sampling matrix*. Since CS only requires that enough samples are received for decoding (i.e., the samples are interchangable), the watermark will be able to be detected as long as enough correct samples are received and the magnitude of the errors is bounded and small [3].

We then find $\hat{\mathbf{x}}$ by solving the optimization problem

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & ||\mathbf{x}||_1 \\ \text{subject to:} \quad & ||\mathbf{y} - A\mathbf{x}||_2^2 < \epsilon, \end{aligned} \tag{3}$$

which can be shown to be equivalent to finding the "sparsest" vector $\mathbf{x}$ that matches the received signal vector $\hat{\mathbf{y}}$ within some error tolerance ϵ.

Finally, since the recovered $\hat{\mathbf{x}} \in \mathbb{R}$ will be continuous-valued while the original message only takes values $x_i \in \{-1, 0, 1\}$, we will need to quantize $\hat{\mathbf{x}}$. Letting $\max(\mathbf{x}, K)$ be the set of x_i with the largest K absolute values, we set $\hat{x_i^{(q)}} = \text{sgn}(x_i)$ for $x_i \in \max(\mathbf{x}, K)$ and $\hat{x}_i^{(q)} = 0$ otherwise.

We measure the MSE between the quantized recovered message and original message

$$\text{MSE} = \frac{\|\mathbf{x} - \mathbf{x}^{(q)}\|_2^2}{K}. \tag{4}$$

From this definition, it is clear that $0 \leq \text{MSE} \leq 4$. We note that typical values range from $0 \leq \text{MSE} \leq 2$ since $\text{MSE} = 4 \Rightarrow \mathbf{x}^{(q)} = -\mathbf{x}$, while $\text{MSE} = 2$ likely implies that the recovery chose all incorrect indices as support for our estimated message. Thus we claim to detect the original message $\mathbf{x}$ when $\text{MSE} < 1$.

3.2. Selecting Watermark Parameters

Since this scheme requires the watermark to be restricted to be a K-sparse signal, we need to verify that that restriction does not make the resulting watermark too "easy" for an adversary to guess using brute-force. We wish to design our watermark such that a third party who knows the watermarking scheme but does not have full information about the shared secrets ($\mathbf{A}$ or $\mathbf{x}$) will not be able to generate a "correct" but unauthorized watermark. In the case where the third party only has knowledge of the message, then the matrix $\mathbf{A}$ must be guessed. However, since guessing each element of the sampling matrix A in infeasible[1], we only consider the more tractable case in which the third party has knowledge of the matrix $\mathbf{A}$ but not of the message $\mathbf{x}$.

The total number of messages of length N and sparsity K is $T = 2^K \binom{N}{K}$. Assuming that every message has an equal chance of being generated by the third party, the probability of guessing the exact message is the reciprocal of this quantity. However, since we do not require perfect recovery for detection of the message, we examine the probability of guessing a detectable message. The number of detectable messages is given by

$$R = \sum_{i=\lceil K/2 \rceil}^{K} \sum_{j=0}^{i-\lceil K/2 \rceil} 2^{K-i-j} \binom{K}{i} \binom{K-i}{j} \binom{N-K}{K-i-j} \tag{5}$$

where by convention $\binom{n}{r} = 0$ if $r > n$. Then the probability of generating a detectable message to spoof an unauthenticated watermark is R/T. Probabilities for randomly generating a detectable message is shown in Table 1.

N	K	p
64	4	4.22×10^{-3}
64	16	4.45×10^{-5}
128	4	1.08×10^{-3}
128	16	2.67×10^{-7}
256	4	2.72×10^{-4}
256	16	1.23×10^{-9}

Table 1. Probabilities of randomly generating detectable message by guessing for different values of K and N

This also shows that a number of distinct messages $\{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_P\}$ can be designed for a single matrix $\mathbf{A}$ to generate watermarks for P different authentic sources without being detected as a different authentic message.

4. SIMULATIONS AND RESULTS

4.1. Experiments

We use the classic 512×512 grayscale Lenna test image [11] for our simulations, and gradient projection for sparse reconstruction (GPSR) [12] to solve the convex optimization problem. We chose a packet size of 128 bytes and inserted a RST marker every 64 macroblocks which corresponds to one at the end of each horizontal row. We assume the JPEG header is transmitted intact.

Figure 1 shows a typical watermarked image and a received image at $\text{PER} = 0.18$. For this work, we assume a Gaussian packet erasure channel. We see the structure of the errors from the lost packets in 1(b). Note that even in the presence of relatively high PER, there is some visually significant information present in the transmitted image that may be worth verifying.

Figure 2 shows the watermark recovery step performed on a typical instance of a received image. The recovered $\mathbf{x}$ is continuous-valued but sparse with the correct support, so the quantization to $\mathbf{x}_q$ still matches the original message exactly and is detectable.

In our simulations, we examined the effects of several parameters of the detection rate of the watermark across varying PER. These parameters included sparsity level N/K, expansion level L/N, and number of quantization steps Q_n. The

[1] While our preliminary studies have shown that some information can be gained if an attacker can correctly guess 80% of the sampling matrix elements within ϵ, for even our smallest case, the sampling matrix A is a real valued 64×256 matrix. This would require an attacker to correctly guess at least 13,108 real valued elements, which is clearly not practical.

(a) (b)

Fig. 1. (a) shows the watermarked image with no transmission errors; (b) shows the received image at $\mathrm{PER} = 0.18$.

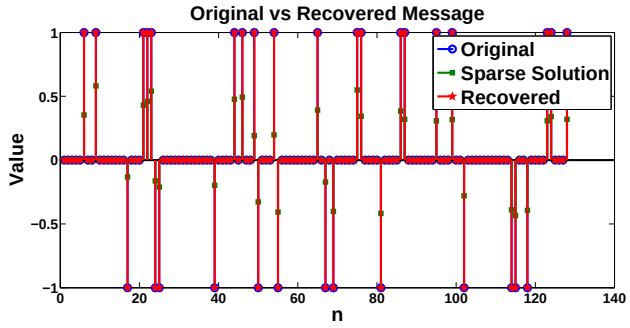

Fig. 2. Recovery step

detection rate for each set of parameters was calculated as the ratio between the number of times the true watermark was detected and the total number of trials. Each setting was run with 100 Monte Carlo trials.

Figure 4 shows that the detection rates across all packet error rates can be increased by increasing the expansion level L/N. Figure 3 similarly shows that detection rates across packet error rates can be increased by increasing the sparsity N/K of the message $\mathbf{x}$. Comparing this result to the analysis from Table 1, we see that there is a tradeoff between security and error protecting performance in designing the sparsity of the message.

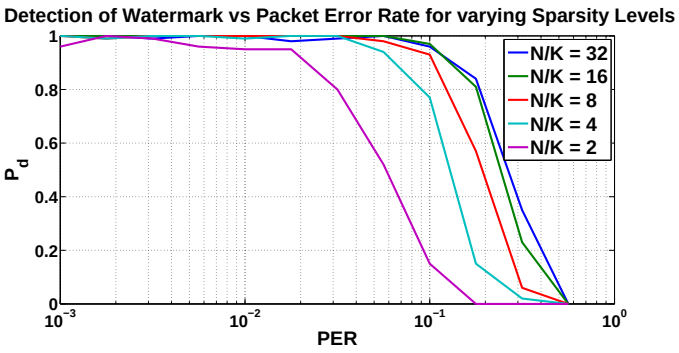

Fig. 3. Detection rate across packet error rates at varying sparsity levels N/K

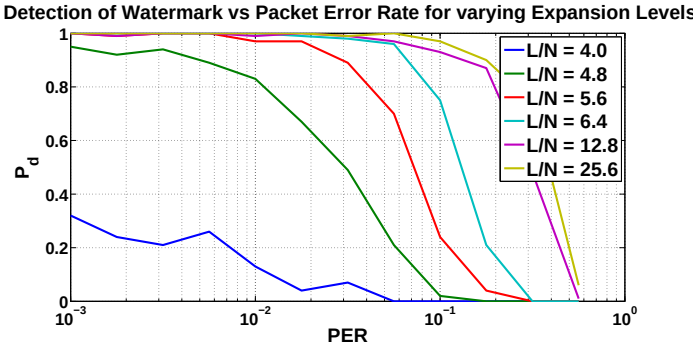

Fig. 4. Detection rate across packet error rates at varying expansion levels L/N

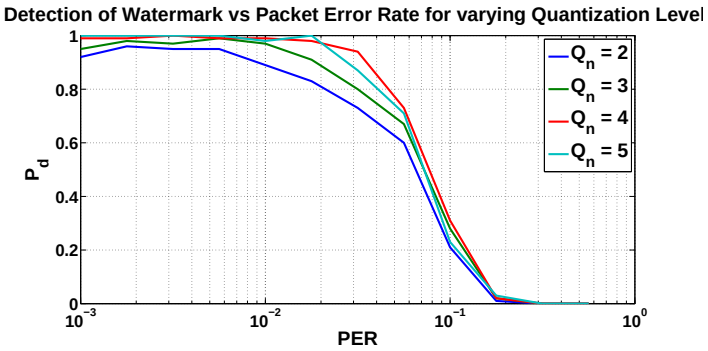

Fig. 5. Detection rate across packet error rates at varying quantization levels Q_n

Figure 5 demonstrates increasing the quantization levels used to embed the watermark can also increase performance. However, there seems to be a point at which this increase no longer results in significant increase in detection rate. This could be due to the fact that increasing the quantization levels increases the magnitude of the gross errors.

5. CONCLUSION

In this work we introduced the application of CS to robust image watermarking. This was done by introducing a watermarking scheme for JPEG encoded images and showing that it was effective in noisy communication channels. Our scheme takes advantage of sparse recovery ideas borrowed from compressed sensing to achieve performance in very noisy conditions. We demonstrated this performance under certain parameters and analyzed the security of the authentication provided by the scheme. Future work includes extending our scheme to video watermarking. We believe our scheme is particularly well-suited for streaming video, since our protocol can handle discarded packets and does not rely on knowledge of the original image. In addition, alternate embedding techniques and types of attacks could be studied.

6. REFERENCES

[1] F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1079–1107, 1999.

[2] D. Donoho, "Compressed Sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[3] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[4] K. Gao, S. N. Batalama, and D. A. Pados, "Compressive sampling with generalized polygons," *IEEE Transactions on Signal Processing*, submitted Nov. 2010.

[5] "Digital Compression and Coding of Continuous-Tone Still Images - Requirements and Guidelines," ITU-T Recommendation T.81, 1992.

[6] M. Gkizeli, D. Pados, and M. Medley, "Optimal signature design for spread-spectrum steganography," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 391–405, 2007.

[7] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673–1687, 1997.

[8] S. Pudlewski and T. Melodia, "Compressive Video Streaming: Design and Rate-Energy-Distortion Analysis," *IEEE Transactions on Multimedia*, In Press 2013.

[9] ——, "A tutorial on encoding and wireless transmission of compressively sampled videos," *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 754–767, 2013.

[10] ——, "DMRC: Distortion-minimizing Rate Control for Wireless Multimedia Sensor Networks," in *Proc. of IEEE Intl. Conf. on Mobile Ad-hoc and Sensor Systems (MASS)*, Macau S.A.R., P.R. China, October 2009.

[11] USC Signal and Image Processing Institute, http://sipi.usc.edu/database/index.html.

[12] M. A. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586–597, 2007.