

BLIND ERROR-FREE DETECTION OF TRANSFORM-DOMAIN WATERMARKS

Mona A. Sheikh and Richard G. Baraniuk

Electrical & Computer Engineering Department
Rice University
Houston TX 77005–1892

ABSTRACT

In this paper we propose a new blind, error-free detection algorithm for watermarking in transform domains. The detection scheme uses linear decoding techniques from the recent theory of Compressive Sensing (CS), whose central idea is that a small number of non-adaptive linear projections of a sparse signal are sufficient for error-free reconstruction of the original signal. We use the fact that natural images are approximately sparse in the DCT or wavelet basis; with an extra step of sparsification or scaling of the coefficients we can decode both the original image and watermark with zero error, despite not knowing the host image. Besides being error-free, our proposed detection algorithm has low complexity compared to other blind algorithms. It can be extended to any transform-domain watermarking method, and also be used to watermark already compressed images.

Index Terms— Blind Watermark Detection, ℓ_1 -decoding, Image compression, Transform Domain Watermarks, Compressive Sensing

1. INTRODUCTION

Efficient and accurate digital watermarking of multimedia is more critical than ever in the Internet age. In watermarking schemes we embed a signature “watermark” in an image or audio file in order to represent and protect ownership. We require a watermark to be imperceptible yet robust against attacks from media pirates who may attempt to remove them. In order to verify ownership we design detectors that can confidently establish the presence of a specific watermark. There is a tradeoff in choosing a simple low-complexity watermark embedding / detection algorithm: making it too simple will also make it susceptible to detection by a third party. Besides copyright protection, watermarking can also be used for content verification, so as to prevent the propagation of illegal media.

We often use specific watermarking schemes to cater to application-specific requirements. But one overarching theme in all applications is accurate detection of the watermark. Here we present a detection scheme that not only detects the

presence of the watermark with 100% accuracy (under specified conditions), but does so with zero error in *estimating* it. Therefore proof of ownership is rightfully established with no room for false positive speculation that a competitor watermark may have been erroneously detected.

The central idea of our scheme stems from the idea of error-free reconstruction of sparse signals by ℓ_1 -decoding in the Compressive Sensing literature [1]. Through ℓ_1 -decoding we can exactly recover a given input signal as long as it is only sparsely corrupted by error. As an extension it has also been shown that the error vector need not be sparse per se for perfect recovery; if it can be made *compressible* (its coefficients decay by a power law), then it will have a good sparse approximation and also can be recovered with small bounded error. As a result, the probability of error-free compressible signal recovery will be slightly compromised compared to sparse signal recovery. In ℓ_1 -decoding, the recovery algorithm does not require any information about the input signal; interpreting this in watermarking terms, the decoding procedure is *blind* to the host image.

There is no extra work in designing a specific watermark or embedding procedure for our detection scheme to work. In fact, it is compatible with any transform domain (DFT, DCT, DWT, etc.) technique for watermark embedding. We use the fact that natural images can be sparsified in some transform basis; i.e., they can be approximately represented by a small number of transform coefficients. This sparsified representation is obtained by thresholding coefficients below a certain value. The watermark is simply added to the sparsified image, as would be done in any transform domain watermarking technique. Alternatively, instead of completely discarding coefficients, we might scale them by a power-law decay and still recover the original watermark within some bounded margin of error. On the detection side, our ℓ_1 -decoding detection technique is applied to the image in its transform basis.

The idea of sparsification of the image is the same as is used in image compression, by JPEG or JPEG2000 for instance. Therefore our proposed watermarking scheme naturally lends itself to watermarking of *compressed* images. There exist other watermarking schemes for compressed images, but the ones we have encountered require knowledge of the host image [2]. Other blind watermarking schemes appear

to have drawbacks like higher error-rates, complex decoding due to host image modeling in an effort to be “informed”, or statistical requirements imposed on the host image [3; 4].

2. COMPRESSIVE SENSING AND ℓ_1 -DECODING

The recent theory of *Compressive Sensing* introduced by Candès, Romberg, and Tao [5] and Donoho [6] demonstrates that a length N signal that is K -sparse in one basis (the *sparsity basis*) can be recovered from $O(K \log(N/K))$ nonadaptive linear projections onto a second basis (the *measurement basis*), that is incoherent with the first.

The recovery of the sparse set of significant coefficients $\{\theta(n)\}$ can be achieved by *optimally* searching for the signal with ℓ_0 -sparsest (the ℓ_0 “norm” $\|\theta\|_0$ counts the nonzero entries in the vector θ) coefficients that agree with the linear measurements made. Unfortunately solving this ℓ_0 optimization problem is prohibitively complex. The practical revelation supporting CS is that it is not necessary to solve the ℓ_0 -minimization problem to recover $\{\theta(n)\}$; a much easier problem yields an equivalent solution (thanks again to the incoherency of the bases); we need only solve for the ℓ_1 -sparsest coefficients θ that agree with the measurements y [5; 6].

CS theory requires that the measurement matrix satisfy a certain restricted isometry property (RIP) to be incoherent with the sparsity basis: an essential condition for error-free signal recovery. Fortunately, matrices whose entries are formed from random sampling of a Gaussian distribution satisfy the RIP. We use Gaussian matrices extensively to obtain linear measurements because of this characteristic [1].

Candès and Tao solve the noisy channel linear decoding problem by using the sparse reconstruction technique in CS. In this problem we want to recover a given message vector f in R^n from corrupted measurements $y = Af + e$, where A is an $m \times n$ matrix ($m > n$) and e is a sparse vector of errors. We can recover f perfectly provided the vector e is sparse enough, and therefore only a small number of elements in y are corrupted by it. The ℓ_1 -decoding attack is to accurately reconstruct the sparse vector e , and hence recover the target message f from $f = A^{-1}(y - e)$ since $A_{m \times n}$ has full column rank [1].

The Candès-Tao ℓ_1 -decoding solution for recovering e is as follows: (1) construct a matrix F such that $FA = 0$, (2) apply F to the corrupted vector y , (3) from the resultant $\tilde{y} = F(Af + e) = Fe$, we are faced with the familiar issue of reconstructing sparse e from its linear measurements y . Besides sparsity or compressibility of e , the *encoding ratio* $m:n$ also affects the probability of error-free detection. The larger the encoding ratio between codeword and message, the lower is the threshold sparsity required for error-free detection [1]. This decoding approach is the main principle of our detection algorithm: the message f is the watermark that needs to be preserved in the sparse image e in the above model.

3. WATERMARKING SCHEME

We can draw parallels between the Candès-Tao approach to decoding sparsely corrupted signals and detection of watermarks buried in sparse image coefficients. In our proposed watermarking scheme we embed a given confidential watermark f by first encoding it as a sequence $p = Af$ and then adding it to sparse image coefficients e ; this will give us our watermarked image coefficients $y = p + e$. This setup is perfect for ℓ_1 -decoding to accurately determine the sparse coefficients e , thereby also deciphering the watermark f .

3.1. Embedding

It is essential to have a sparse (or compressible) representation of the image before the watermark can be added to it. Transform domains like DFT, DCT and DWT are useful in this regard, since images can be represented by a few large-valued transform coefficients. This fact is used in image compression where small-valued coefficients below some threshold value are discarded. In this sense images have a sparse representation in these transform domains.

Transform-domain watermarking techniques have been shown to be more robust and tamper-proof as compared to straightforward spatial watermarking [7; 8]. Transform techniques embed the watermark in the most critical features of the image, so that fiddling with the image to un-watermark it will cause image quality to deteriorate as well. Each transform domain has its own virtues, the most popular ones being Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) that are used in the JPEG and JPEG2000 image compression standards, respectively.

Our proposed algorithm also adds the watermark in the transform domain. But we introduce one extra step of either sparsifying (to make sparse) or scaling down (to make compressible) the image coefficients before adding the encoded watermark. To simplify terminology we hereon refer to the ratio of total number of coefficients to non-sparsified/scaled coefficients as the *compression ratio*; i.e., a compression ratio of 3:1 means that a third of the coefficients have been scaled/sparsified.

We use a gaussian matrix $A_{m \times n}$ (whose entries are generated from a secure seed known only to the embedder and detector) to encode the confidential watermark. By doing so we have introduced implicit security in the watermark detection algorithm since the detector requires the secure seed to generate A , which is key for ℓ_1 -decoding to work.

In the simulations in this paper we consider DCT-domain watermarking, but the embedding / detection procedure described remains the same for any transform domain watermarking method. In the DCT domain we choose to embed the watermark in the important middle band frequencies; embedding in the low frequency coefficients (smooth regions in image) makes the watermark perceptible, while the high frequency coefficients are prone to noise, filtering and lossy

compression. For instance, in JPEG compression the higher frequency coefficients are typically what are eliminated after quantization, leaving our watermarked midband coefficients intact. In the DWT domain we would have chosen to embed the watermark in the high resolution bands, while in DFT we would embed in the phase component of the image.

3.2. Detection

The detection procedure involves first transforming the image back to the transform domain and then performing ℓ_1 -decoding to obtain the original watermark.

In summary our watermarking recipe is:

Embedding: (1) Transform the image to appropriate domain (2) Sparsify or scale the transform coefficients (3) Encode confidential watermark using encoding matrix A generated by seed (4) Add encoded watermark to transform coefficients. (5) Inverse transform to get watermarked image.

Detection: (1) Transform given image to appropriate domain (2) Recover sparse image coefficients by using ℓ_1 -decoding with given A . (3) Recover confidential watermark f by linear operation $f = A^{-1}(y - e)$.

3.3. Factors affecting detection

One caveat in ℓ_1 -decoding is that the image must have a certain threshold *compression ratio* to have error-free detection. Hence the compression ratio plays an important role in affecting detection probability. The fewer the image coefficients that are kept, the higher will be the probability of error-free watermark detection. On the other hand, keeping too few image coefficients will affect image quality. This tradeoff should be kept in mind; choosing the lowest compression ratio while still maintaining error-free detection is ideal.

A second factor that affects the ability to detect the watermark error-free is the *encoding ratio* between the encoded watermark and the confidential watermark ($m:n$) as discussed in Section 2. Generating a longer bit sequence to encode a given watermark (using the appropriately sized $A_{m \times n}$) will improve watermark detection probability in an image with a given compression ratio.

If we are scaling the coefficients down (as opposed to sparsifying them), the *nature* of scaling will affect detection probability, given a fixed compression and encoding ratio. We found that scaling so that the coefficients decayed according to a power law: $|\alpha|_k \leq Bk^{-s}$ required the least number of coefficients to be modified [1]. For purpose of uniformity, we assumed $B = \text{norm}(\text{midband coefficients to be scaled})$.

4. SIMULATION RESULTS

We ran simulations to compare the effect that varying compression ratio, watermark encoding dimensions and coefficient scaling (as opposed to sparsifying) have on the probability of error-free watermark detection. We ran 100 itera-

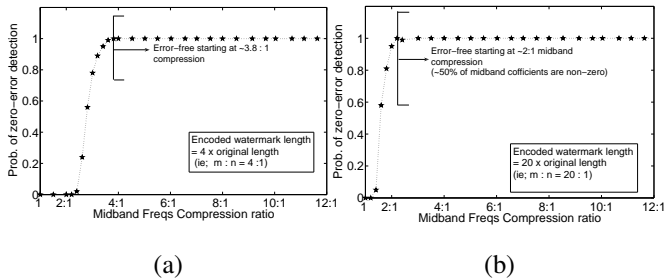


Fig. 1. Detection probability vs. Compression (by sparsity) ratio for encoding ratio $m : n$ (encoded watermark length : original watermark length) of (a) 4 : 1 and (b) 20 : 1.

tions for each compression ratio, generating a new gaussian matrix A in each iteration. Detection during an iteration was deemed successfully error-free if the MSE of the detected watermark was on the order of machine precision (10^{-16} in our case). Frequency of correct detection over the 100 iterations was plotted against the image compression ratio (whether by sparsification or scaling). All simulations used the DCT domain, taking a global DCT of the image to embed the watermark in its midband. But the numerical results shown hold true for any transform-domain method of watermarking.

We see from Figure 1(a) and (b) that the length of the codeword (m) used to encode a watermark of given length (n) will determine the threshold compression ratio at which zero-error detection begins. The longer this codeword is, the easier it is to backsolve and retrieve the original watermark using $f = A^{-1}(y - e)$, since essentially we have an overdetermined system of linear equations as $m > n$. In Figure 1(b) we see that altering the watermark codeword length to be a factor of 20 greater than the source watermark from a factor of 4 improves the threshold compression ratio from approximately 4:1 to just 2:1. In other words, sparsifying half the coefficients (instead of 75% of them) will still enable us to detect the watermark perfectly. In particular, in 8×8 blocking of DCT coefficients, a set of ~ 20 middle frequencies of the 8×8 block is often designated as the “midband” where watermarks are often embedded. Therefore to embed a watermark by our proposed technique only half the coefficients in the midband need to be sparsified before adding the watermark, while still maintaining error-free detection. Sparsifying fewer coefficients also corresponds to an improvement in image quality, albeit very marginal; we make note here that the DCT-embedded watermark was not visually perceptible in any simulation despite the sparsification. This is not surprising since a low quality factor JPEG compression might discard even 70% of image coefficients and still remain indistinguishable to the eye.

Figure 2(a) illustrates detection probability as it is affected by scaling the coefficients instead of sparsifying them. The hit we suffer is that now a minimum of 66% of the coefficients require scaling as opposed to just half of them in Figure 1(b)

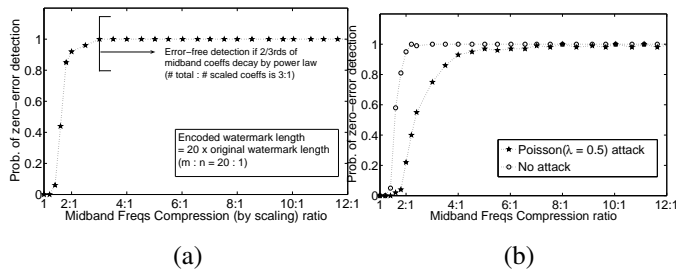


Fig. 2. (a) Detection probability vs. Compression (scaling) ratio after scaling by power law decay with decay constant $s=5$, and (b) Detection probability affected by Poisson noise attack (compression by sparsifying)

for the same $m : n$ encoding ratio.

Lastly, in Figure 2(b) we attack the image with Poisson noise from randomly sampling a Poisson distribution of variance 0.5. It appears that the watermark is now more difficult to detect at low compression ratios, and the higher the variance, even more difficult the error-free detection. Also, our detection scheme is not robust to Gaussian attack; this is a topic of further research (Section 5).

However, we make note that our requirements to qualify as a “successful” detection attempt in this analysis are extremely strict: the detected watermark should be equal to the original within machine precision accuracy. If we were to slightly raise the permissible MSE threshold to qualify as successful detection we would also raise probability of detection success. This would be competitive with other detection algorithms whose main concern is simply detection – not necessarily error-free watermark decoding. On the other hand, it would be prone to false positive and false negative detection.

5. DISCUSSION AND CONCLUSIONS

The watermarking technique that we have proposed has some principal advantages. Despite being a blind scheme (no access to host image) it guarantees watermark detection and estimation accuracy if the image has the required minimum compression. If the image is already compressed, then the watermark can be added immediately; else the image coefficients can be appropriately zeroed out or scaled down to attain the minimum required compression. The confidential watermark is encoded by a Gaussian matrix with a secret seed known only to the detector; in this sense detection is inherently secure against a rogue third party who may want to remove the watermark. Furthermore, our scheme can adopt existent watermark design methods in transform domain watermarking for still improved image quality.

Since the advent of image compression, watermarking schemes have been developed to take advantage of compressed data. Our ℓ_1 -decoding-based watermarking scheme nicely lends itself to this function, since the sparsification

step before adding the watermark is already done. The ℓ_1 -decoding procedure is a linear program with especially low complexity, and is therefore suitable for doing watermark detection on the fly as has been proposed by other compressed image watermarking methods [2]. A potential application for a fast decoding solution might be watermark detection in a digital camera that already stores JPEG-compressed data. The decoding accuracy of the algorithm renders it particularly useful for error-intolerant data-hiding applications in which perfect message decryption is critical.

An important area of investigation for our watermarking algorithm is its robustness against attacks. The choice of encoding matrix A plays a role in surviving noise attacks; we have shown that using a gaussian matrix to encode the watermark is resistive to Poisson noise. It is not known which measurement matrix is optimal; this is an area of ongoing research in CS [9]. We are also concerned with watermark survival after image compression; embedding in midband frequency coefficients (as opposed to higher frequencies) improves, but does not guarantee compression-survival. We plan to explore intelligent watermark coefficient scaling methods to improve on this.

6. REFERENCES

- [1] E. J. Candès and T. Tao, “Decoding by linear programming,” *IEEE Trans. Inform. Theory*, vol. 51, pp. 4203–4215, Dec. 2005.
- [2] P. Su, H. Wang, and C. Kuo, “Digital watermarking on ebcoT compressed images,” *SPIE Conference on Applications of Dig. Image Processing XXII*, 1999.
- [3] H. Malik, A. Khokhar, and R. Ansari, “Improved watermark detection for spread-spectrum based watermarking using ICA,” *Fifth ACM Workshop on DRM*, 2005.
- [4] T. Chen and T. Chen, “A framework for optimal blind watermark detection,” in *ACM Multimedia 2001 Workshop on Multimedia and Security*, Ottawa, Oct. 2001.
- [5] E. J. Candès and T. Tao, “Near optimal signal recovery from random projections: Universal encoding strategies?,” Preprint, 2004.
- [6] D. L. Donoho, “Compressed sensing,” *IEEE Trans. Info. Theory*, vol. 52, no. 4, pp. 1289–1306, September 2006.
- [7] G. Langelaar, I. Setyawan, and R. Lagendijk, “Watermarking digital image and video data: A state-of-the-art overview,” *IEEE Signal Processing Magazine*, 2000.
- [8] B. Furht and D. Kirovski, *Multimedia Security Handbook*, 2005, Boca Raton, FL.
- [9] R. G. Baraniuk, M. Davenport, R. A. DeVore, and M. B. Wakin, “The Johnson-Lindenstrauss Lemma Meets Compressed Sensing,” 2006, Preprint.