

# GA Based Robust Blind Digital Watermarking

V.Álvarez, J.A.Armario, M.D.Frau, F.Gudiel, M.B.Güemes, E.Martín, A.Osuna<sup>1,2</sup>

*Dept. Matemática Aplicada 1 and Álgebra  
Universidad de Sevilla  
Sevilla, Spain*

---

## Abstract

A genetic algorithm based robust blind digital watermarking scheme is presented. The experimental results show that our scheme keeps invisibility, security and robustness more likely than other proposals in the literature, thanks to the GA pre-treatment.

*Keywords:* Watermarking, Genetic Algorithm, Hadamard Transform.

---

## 1 Introduction

Digital watermarking concerns those methods about how to hide a special mark into digital multimedia data to solve the problems of legal ownership, integrity and authenticity of the original data [2].

The techniques proposed so far can be grouped into two different approaches, depending on whether the watermark is embedded into the least significant bits (spatial domain approach, see [7]) or it is embedded attending to the perceptually most significant frequency components of the container

---

<sup>1</sup> All authors are partially supported by the research project FQM-016 from JJAA.

<sup>2</sup> Email: {valvarez,armario,mdfrau,gudiel,bguemes,emartin,aosuna}@us.es

image (frequency domain approach, see [1]). Usually one tends to apply techniques of the second type, since spatial domain approaches have relatively low information hiding capacity and, what is more important, can be easily erased by lossy image compression.

Most of frequency domain approaches use discrete wavelet transform (DWT), discrete Fourier transform (DFT) and discrete cosine transform (DCT). Very recently fast Hadamard transform (FHT) has arisen as a promising alternative [5] and [10].

No matter the processing speed is, watermarking is usually required to muster three conditions: *security*, *imperceptibility* and *robustness*.

Security is concerned with embedding a watermark into a piece of content at an untrusted user device without compromising the security of the watermark key, the watermark or the original [6].

Perceptibility measures whether perceptible artifacts on the watermarked image are introduced, that is, if the presence of the watermark in the final image is noticeable. This magnitude is measured in terms of the *Peak Signal to Noise Ratio*, or *PSNR* in brief. It is most easily defined via the *mean squared error (MSE)*, so that for images with maximum possible pixel value *range* (i.e. 255 or 1 depending on whether byte or real storing method is adopted), *PSNR* is calculated as:

$$(1) \quad PSNR = 10 \log_{10} \frac{range^2}{MSE}.$$

Here, for two  $m \times n$  monochrome images  $K = (k_{i,j})$  and  $L = (l_{i,j})$  (where one of the images is considered a noisy approximation of the other), *MSE* is defined as follows:

$$(2) \quad MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (k_{i,j} - l_{i,j})^2.$$

The robustness of a watermark depends on whether it fails to be detected after unintentional or even malicious transformations [8]. It is usually measured in terms of the *Normalized Correlation (NC)* between the extracted watermark image  $EW = (ew_{i,j})$  (presumably modified) and the original watermark  $W = (w_{i,j})$ ,

$$(3) \quad NC_{W,EW} = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} w_{i,j} ew_{i,j}}{\sqrt{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} w_{i,j}^2} \sqrt{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} ew_{i,j}^2}}.$$

Given a watermarking scheme, it may be straightforwardly improved in terms of security, imperceptibility and robustness by simply introducing some

pretreatment to the watermark image in order to destroy space relativity [9] or [10]. Taking the work in [10] as starting point, in this paper we describe an improved blind (that is, the original cover image is not needed for extracting the watermark) watermarking scheme, with the following advantages:

- There is no dependence on the sizes of the watermark and cover images, in [10] it is forced to be 1/8.
- The trade-off between imperceptibility and robustness is measured in terms of a parameter  $b$ . The greater  $b$  is, the nearer  $NC$  is to 1, the smaller  $PSNR$  is. Accordingly, the smaller  $b$  is, the smaller  $NC$  is, the greater  $PSNR$  is. Some explanations (beyond simple computational evidence!) will be given in order to justify the optimal value for  $b$ , depending on the way in which the image is being stored (real or byte representation).
- Robustness, security and imperceptibility of the scheme are significantly improved thanks to a pretreatment of the watermark image. We have designed a genetic algorithm (in the sequel, GA in brief), looking for a permutation of the original watermark which is as uncorrelated as possible to it. This GA, equipped with a specific crossover operator specially designed for the occasion, beats usual GA equipped with classical crossover operators concerning permutations problems (such as *order 1*, *partially mapped* and *cycle* crossovers, or *edge recombination*).

## 2 GA for Uncorrelated Permuted Images

Given an image  $Im$ , we want to find a permuted image  $PIm$  of  $Im$ , so that their normalized correlation  $NC(Im, PIm)$  is as less as possible. In what follows, we assume that the reader is familiar with the general framework of GAs, and their usual elements and characteristics. If necessary, [3] (and the references there included) is a good place to get a general overview of the subject.

Here we use a *Steady-State* GA, in the sense that three offspring (two coming from crossover, and one more coming from mutation) are generated per generation, which will replace the worst adapted individuals at the moment. This way, we use also *elitism*, since we always keep the fittest solution so far. Additionally, we use a “no duplicates” policy, in the sense that identical individuals are not allowed to occur in the same generation.

In order to select two individuals for reproduction purposes, we use *rank-based* selection, by means of *linear ranking*. More concretely, assume that the size of the population is  $\mu$ . Then sort the population in terms of fitness, so

that fittest has rank  $\mu$  and worst rank 1. Now fix a factor  $1 \leq s \leq 2$  (we use  $s = 1.5$  in the sequel). In these circumstances, the probability that the  $i^{th}$  individual is selected for reproduction is given by


$$(4) \quad P_i = \frac{2-s}{\mu} + \frac{2(i-1)(s-1)}{\mu(\mu-1)}.$$

Depending on whether the factor  $s$  is closer to 1, fitness is accordingly relativized for choosing the individual.

Our population consists of 20 individuals, and every run is limited to 50 generations. The fitness function consists in the normalized correlation (3) between the permuted image and the original watermark, so that the lesser  $NC$  is, the fitter an individual is.

Since our watermark images are binary  $m \times m$  matrices, we can easily encode them as  $m^2$ -length binary vectors. Furthermore, we can just save the positions in which 1 (analogously, 0) entries are displayed. Assume that the watermark image consists of  $k$  white pixels. Then the set of its permuted images is uniquely determined by the set of  $k$ -subsets of  $\{1, \dots, m^2\}$ .

Given two such different  $k$ -subsets  $S_1$  and  $S_2$ , we select proportionally positions in  $S_1$  and  $S_2$  attending to their fitness. Assume that  $S_1$  has better fitness than  $S_2$ . Fix randomly a real number  $0.5 \leq r \leq 1$ . Then we will get a  $\lfloor k \cdot r \rfloor$ -subset  $S$  of  $S_1$ , and join this subset with a random  $k - \lfloor k \cdot r \rfloor$ -subset of  $S_2 - S$ . Calculations show that this custom-made crossover operation performs better than alternative traditional ordering crossovers. More concretely, the GA

outputs  $NC_1 = 0.7952$  and  $NC_2 = 0.1159$  progressing from the images 

and  (for which the initial  $NC$  values are 0.799 and 0.156, respectively).

In the following section we describe a watermarking scheme. We claim that pretreatment of the original watermark (so that a minimally correlated permuted image is obtained and used instead), improves the watermarking scheme in an obvious way, not only from the security point of view (no matter one knows the extracting procedure, the extracted watermark will be meaningless), but also from the point of view of invisibility, without loose of robustness.

### 3 The watermarking scheme

Let  $A$  be the original cover grayscale digital image, encoded as a  $n \times n$  matrix with integer entries in  $\{0, \dots, 255\}$ . Let  $W$  be the binary watermark, encoded as a  $m \times m$  matrix with 0, 1 entries. The embedding procedure may be detailed as follows:

- Find a normalized Hadamard matrix  $H$  of size  $4t$  closest to  $\lfloor \frac{n}{m} \rfloor$ .
- Partition the original image  $A$  into non-overlapped blocks of size  $\lfloor \frac{n}{m} \rfloor$ . Consider the sub-blocks of size  $4t \times 4t$  naturally embedded, which we denote by  $A_i$ ,  $1 \leq i \leq m^2$ .
- Apply the *extended* Hadamard Transform to  $A_i$ , in order to obtain

$$(5) \quad B_i = \frac{HA_iH^T}{4t}.$$

- Select two entries  $b_1$  and  $b_2$  in  $B_i$  in the same row (or column), say  $b_1 = B_i(3,3)$  and  $b_2 = B_i(3,5)$  for instance. Depending on whether the corresponding pixel  $i$  in  $W$  is 0 or 1, force that  $b_2 > b_1$  or  $b_2 < b_1$  accordingly. To this end, fix a value  $b$ , and take  $d = \frac{|b_1 - b_2|}{2}$ . Then set:
  - If  $i = 0$  and  $b_2 \leq b_1$  then actualize  $b_1^* = b_1 - d - b$ ,  $b_2^* = b_2 + d + b$ .
  - If  $i = 1$  and  $b_2 \geq b_1$  then actualize  $b_1^* = b_1 + d + b$ ,  $b_2^* = b_2 - d - b$ .
- The watermarked block  $A_i^*$  is obtained by the inverse transform of (5),

$$(6) \quad A_i^* = \frac{H^T B_i^* H}{4t}.$$

At this point, we would like to make two major comments:

- (i) Taking a deeper insight in the Hadamard transform (5) one deduces that a change  $\Delta$  in  $b_i$  translates into a change about  $\lceil \frac{\Delta}{2t} \rceil$  in  $A_i^* = \frac{H^T B_i^* H}{4t}$ . Hence, in order to get noticeable byte changes, we should take  $b \approx t$ . In the case of real encoded images,  $b$  may be chosen arbitrarily small (at the risk of decreasing the normalized correlation of the extracted watermark).
- (ii) Although the *Hadamard Conjecture* about the existence of these matrices in every order  $4t$  remains open, there are well known families of Hadamard matrices filling an infinite amount of sizes  $4t$  [4].

The extraction procedure is just the inverse procedure of embedding.

Let  $A_i^*$  be the  $i^{th}$ -block of the watermarked image. In order to recover the pixel  $i$  of the watermark one must simply proceed as follows:

- By (5), form  $B_i^* = \frac{HA_i^*H^T}{4t}$ .
- Let  $b_1$  and  $b_2$  be the entries used in the embedding procedure. If  $b_2 > b_1$  set  $i = 0$ , and  $i = 1$  otherwise.

## 4 Experimental results

We have used 5 typical benchmark  $512 \times 512$  cover images (Lena, Baboon, Boats, Peppers and Testlena) and the two precedent  $64 \times 64$  images as un-

derlying watermarks. We have fixed  $H$  to be the  $8 \times 8$  Sylvester Hadamard matrix. Calculations show that the proposed watermarking scheme is robust under different attacks, such as jpeg compression (with quality factors 80% and 90%), Gaussian noise (of mean 0 and variance 0.001) and salt-and-pepper noise (of density 0.01). Although the proposed watermarking scheme works fine without any need of pretreatment of the watermark, we want to emphasize that permuting the initial watermark ensures that security and imperceptibility (and even robustness to a somewhat lesser degree) are enhanced.

## References

- [1] Cox, I.J., J. Kilian, F.T. Leighton and T. Shannon, *Secure spread spectrum watermarking for multimedia*, IEEE Trans. Image Proc. **8** (1997), 1673–1687.
- [2] Cox, I.J., M.L. Miller and J.A. Bloom, “Digital watermarking”, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.
- [3] Eiben, A.E. and J.E. Smith, “Introduction to Evolutionary Computing”, Springer, Natural Computing Series, 2003.
- [4] Horadam, K.J., “Hadamard matrices and their applications”, Princeton University Press, Princeton, 2007.
- [5] Ho, A.T.S., J. Shen, S.H. Tan and A.C. Kot, *Digital image-in-image watermarking for copyright protection of satellite images using the fast Hadamard transform*, in Proceedings of the 24th IEEE Int. Geoscience and Remote Sensing Symposium, Toronto, Canada (2002), 3311–3313.
- [6] Lemma, A., S. Katzenbeisser, M. Celik and M. van der Veen, *Secure Watermark Embedding Through Partial Encryption*, in Proceedings IWDW2006, Eds. Y.Q. Shi and B. Jeon, LNCS **4283**, Springer Verlag, Berlin (2006), 433–445.
- [7] van Schyndel, R.G., A.Z. Tirkel and C.F. Osborne, *A digital watermark*, in Proceedings IEEE Int. Conf. Image Processing **2** (1994), 86–90.
- [8] Voloshynovskiy, S., F. Deguillaume, O. Koval and T. Pun, *Robust watermarking with channel state estimation*, Signal Processing, special issue on Security of Data Hiding Technologies, Eds. S. Voloshynovskiy, T. Pun, J. Fridrich, F. Pérez-González, N. Memon, **83**, 10, (2003).
- [9] Zhang, D., J. Xu, H. Li and H. Li, *A novel image watermarking algorithm with fast processing speed*, in Proceedings ICIECS 2009, IEEE Press, China, (2009).
- [10] Zhang, Y., Z.M. Lu and D.N. Zhao, *A blind image watermarking scheme using fast Hadamard transform*, Information Technology Journal **9** (2010), 1369–1375.