Multi-Objective Genetic Algorithm for Semi-Fragile Watermarking

D. Sal, M. Graña

Dept. CCIA, UPV/EHU, Spain

Abstract

We propose an evolutionary algorithm for the enhancement of digital semi-fragile watermaking based on the manipulation of the discrete cosine transform (DCT). The algorithm searches for the optimal localization in the image DCT to place the watermark image. The problem is stated as a multi-objective optimization problem that involves the simultaneous minimization of distortion and robustness criteria.

Keywords: Genetic Algorithm, Watermarking, Multi-objective Optimization.

1. Introduction

Nowadays, thanks to the advance of the computer technology (hardware and software) and to the improvements in the data distribution nets, we have many facilities to manipulate, distribute and copy digital images. A growing concern is the problem of ensuring the property and integrity of the images. In this situation, some technique that allow us to protect our products against accidental or deliberate attacks is highly desirable. Watermarking is one them [6], [2], [7], [7].

A watermark is a signal carrying some specific information (e.g. about the owner) that can be embedded in an image in an imperceptible way and be recovered later.

A robust mark can demonstrate the authorship of an image (if two persons claim it and one of them recovers a mark that identifies him, the dispute could be solved) [1], [7]. A semi-fragile mark can demonstrate that the image has been manipulated (if the recovered mark is corrupt) [6], [1]. We will focused on this last case.

Our mark must be corrupted when the image is modified. Nevertheless, it could be interesting that the mark was robust to operations as filtering and compression to be able to distribute it. The standard lossy compression technique JPEG works in the coefficients of the image transform. The DCT has been the classical choice for this operation. Therefore we will work on the DCT image transform.

On the other hand, a watermarked image must be as indistinguishable of the original one as possible. The watermarking process must introduce the minimum possible visual distortion in the image.

These two requirements are the objectives of our work and can be contradicting in some instances. The trivial watermarking approach consists in the addition or substitution of the watermark in the image transform high frequency coefficients. That way, the distortion is perceptually minimal, because the watermark is embedded in the noisy components of the image. However, this approach is not robust against smoothing and lossy compression. The robustness can be enhanced putting the watermark in other regions of the image transform, at the cost of increased distortion. Combined optimization of the distortion and the robustness poses a multi-objective optimization.

The Pareto-Front is the set of non-dominated solutions. A non-dominated solution is not improved in all the components of the vectorial objective function by any of its neighboring solutions [6].

The problem shows the typical combinatorial explosion: the number of possible solutions is the number of combinations of the set of image pixel positions over the size of the mark to be placed. We define an evolutive strategy that tries to provide a sample of the Pareto-Front preserving as much as possible the diversity of the solutions.

2. Algorithm

We have an image X of size m_x x n_x that we want to protect. To do that, we use a watermark W of size m_w x n_w. Actually, our mark is a small image or logo. So, the corruption of the recovered mark is detected by visual inspection, and can be measured by correlation with the original mark.

The DCT of the image and the mark are denoted X_t and W_t respectively.

Given two coordinates *k, l* of *W*, $1\leq k\leq m_w$, $1\leq l\leq n_w$, we denote *x(k,l), y(k,l)* the coordinates of X_t where the coefficient *W(k,l)* is added to embed the mark.

2.1. Multi-Objective Fitness

A solution *S* is represented as a m_w x n_w matrix in which every position $W_t(k,l)$ takes two positive values: *x(k,l)* and *y(k,l)*.

Two fitness functions f_1 and f_2, measure the robustness and distortion of the mark placement, respectively. They compose the vector objective function to be optimized.

Robustness fitness function f_1: Robustness refers to the property of recovering the mark even when the watermarked image has been manipulated. We are interested in having robustness against lossy compression and smoothing. Both transformations affect the high and preserve the low frequency transform coefficients. Therefore the closer to the origin of the transform space the mark is located, the higher the robustness of the mark. Besides, as the watermark is really a DCT image transform, it is necessary to take into account that its main information lies in the low frequency coefficients. So they must have higher priority than the low frequency coefficients to be embedded in the positions that are nearer to the low frequencies of X_t. The robustness fitness function has the following form:

$$f_1 = \sum_{k=1}^{m_w}\sum_{l=1}^{n_w} \sqrt[\alpha]{x(k,l)^2 + y(k,l)^2 + k + l} \qquad (1)$$

$$\alpha = F \cdot \frac{\sqrt{x(k,l)^2 + k} + \sqrt{y(k,l)^2 + l}}{x(k,l) + y(k,l)} + d$$

The function in eq. 1 possesses the following properties:

As the position in X_t where *S(k,l)* is embedded approaches the low frequencies, the function value decreases smoothly.

As the pixel of W_t is more important (nearest to the W_t low frequencies), the exponential value of α increases smoothly.

Thus, to maximize the robustness, f_1 must be minimized.

Distortion fitness function f_2: The true distortion is computed as the squared difference between the original image and the inverse of the marked DCT. Final results in the figures will reflect this value. However, to avoid the computational cost of the DCT inversion, we use as the distortion fitness function of the evolutionary algorithm an approximation based on the observation that the distortion introduced adding something to a DCT coefficient is proportional to the absolute value of that coefficient. Thus, the distortion fitness function to be minimized is the following one:

$$f_2 = \sum_{k=1}^{m_w}\sum_{l=1}^{n_w} \left| Xt(x(k,l), y(k,l)) \right| \qquad (3)$$

2.2. Operators

A population *P* contains P_s individuals, which are solutions to the problem. We denote *Of* the offspring population.

Selection Operator: This operator generates *Of* from *P*. To do this, *P* has previously been ordered according to its range and distance between solutions proposed in [1]. The selection is realized choosing stochastically the individuals, but with more probability to select the individuals that are at the beginning of the sorted list.

Merge operator: This operator is applied with P_c probability and is used to recombine each couple of individuals and obtain a new one. Two points from the solution matrix are stochastically selected as cut points, and the individuals are recombined as shown in the next figure.

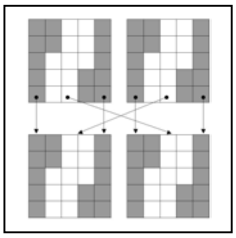

Fig1: Merge Operator 2 cut points based.

Mutation operator: Every element of an individual solution *S* undergoes a mutation with probability P_m. The mutation of an element consist of displacing it to a position belonging to the 8-Neighborhood: given a pixel $W_t(k, l)$ located in the position *x(k,l), y(k,l)* of X_t, the new placement of $S(k,l) \in \{ X_t(x(k,l)\pm1, y(k,l)\pm1) \}$. The direction of the displacement is chosen stochastically. If the selected position is out of the image, or collides with another assignment, a new direction is chosen.

Reduction operator: At this moment there are two populations: *P* and *Of*. Now it is time to select the individuals who are going to form the new one. Both populations are joined in a new one of size $2{\cdot}P_s$. This population is sorted according to the *fast_non_dominated_sort* proposed in [1]. This ensures an elitist selection. The new population *P* is composed of the best P_s individuals according this sort.

2.3. Algorithm

The first step of the GA is the generation of an initial population P and the evaluation of each individual's fitness. Once done this, the habitual curl begins.

An offspring population *Of* is calculated by means of the selection, merge, and mutation operators. The new individuals are evaluated before joining them to the population P. Finally, through the selection operator, the population P for the next iteration is ready.

Since the GA works with many possible solutions, it is probably that several non-dominated solutions were found. Thus, the stop criterion verifies the whole population and finishes the algorithm when no one has improved in n consecutive iterations. The improvement measure is *crowded_comparison* taken from [1]. We divide the image transform into many overlapping regular partitions. A initialization on each partition are executed, allowing the algorithm to evolve without restricting it to the initial partition. In each initialization, a GA is executed to obtain a set of non-dominated solutions. The Pareto-Front is the union of non-dominated solutions from all initializations. A pseudo-code for de GA is the following:

```
begin
    For each quadrant
        P = Generate_Initial_Population();
        Fitness_Functions_Evaluation();
        While Stop < n
            Couples = Selection_Operator(P);
            Of = Merge_Operator(Couples);
            Mutation_Operator(Of);
            F = Join(P,Of);
            P = Reduction_Operator(F);
            For each individual in P
                If crowded_comparison(this, best)
                    Stop = 0; Best_population = P;
                Else
                    Stop++;
    Pareto = Pareto _ non_dominated_solutions(P)
Pareto = non_dominated_solutions(Pareto).
```

Once finished the process and chosen a solution, the mark is embedded adding its coefficient to the coefficients of X_t according to mapping specified in S. Before de coefficients are added, they are multiplied by a small value to minimize the impact.

By knowing the positions where the mark has been embedded, the original image and the original mark, it is possible to recover the mark easily.

3. Results

The algorithm has been applied over an image of size 400 x 500. X_t has been divided in 272 overlapped and regular quadrants of size 100 x 100. In each initialization the watermark has been stochastically distributed through the corresponding quadrant. As watermark has been used a logo of size 32 x 32. The GA was executed with $P_s = 20$, $P_m = 0.05$, $P_c = 0.9$.

For comparison purposes the problem has been solved by means of a local search with the same parameter values. Figure 2 shows the Pareto-Front found with both algorithms. The GA has found 359 non-dominated solutions while the local search only 62.

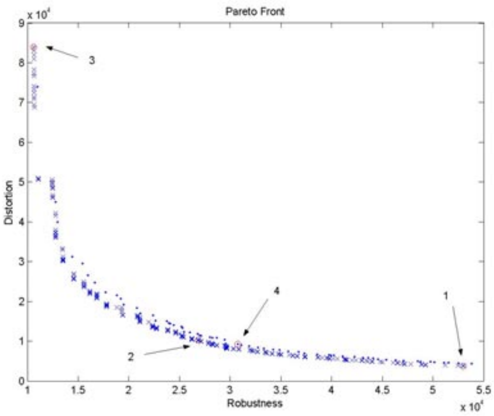

Fig 2. Pareto-Front founded by GA 'x' and local search '·'

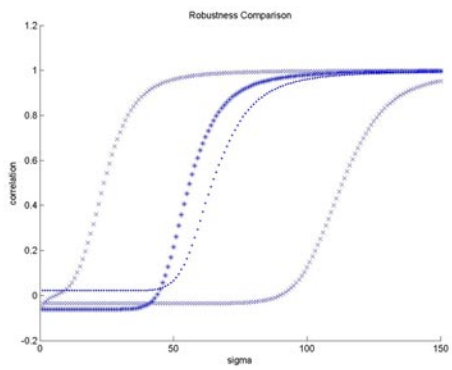

Fig 3: Robustness comparison by means of the correlation coefficient. Solutions 1 and 3 are represented by 'x'. Solution 2 by '*' and Solution 4 by '·'.

The circles in Figure 2 represent different solutions. The first circle has the best distortion value, but the worst robustness value. The solution 3 is the opposite case.

Figure 4 shows the visual distortion effects of each solution, and Figure 3 plots the correlation coefficient between the original watermark and the watermark

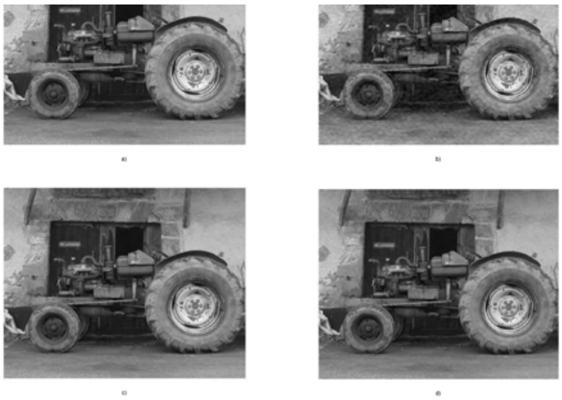

Fig 4. a) Original Image b) Image marked with solution 3, c) Solution 2, d) Solution 1.

corrupted by a low-pass filter with increasing filter radius sigma. This plot shows how the watermark solution 2 obtains a good correlation coefficient much earlier than solution 1 (note in figure 2 scarcely there are any differences between both images). It can be appreciated also how the robustness is higher in the solution 2 (GA) than in the solution 4 (Local Search), Figure 5 shows graphically the corruption effects on the recovered mark from solution 2.

4. Conclusions

We present an evolutionary algorithm to find an optimal watermark's placement in an image to protect it against undesirable manipulations. It is desirable that the watermark remains recognizable when the image is compressed or low-pass filtered. We state the problem as a multiobjective optimization problem, with two fitness functions to minimize. The algorithm tries to obtain the Pareto-Front to find the best trade-off between distortion of the original image in the embedding process and robustness of the mark.

The main inconvenient of the method is that to recover the watermark, it is necessary the original image, the original mark and the positions where the mark was placed. Moreover, the algorithm is still not appropriate for applications with strong restrictions of time due to the fact that to form the Pareto-Front, looking for many solutions is necessary. For example, in figure 2, the algorithm worked with 5.440 possible solutions to find the 189 solutions that form the front. Once the desirable solution is chosen from the Pareto-Front, the embed process is inmediate.

5. References

[1] K. Al_Sultan, M.Saeb, U.Badawi "A proposed semi-fragile watermarking scheme for image authentication".

[2] D. Augot, J.M. Boucqueau, J.F. Delaigle, C. Fontaine and E. Goray (1999) "Secure delivery of images over open networks", Proceedings of the IEEE, vol. 87, Nº7:1251-1266.

[3] T. Bäck, D. B. Fogel and T. Michalewicz(2000) "Evolutionary Computation1. Basic Algorithms and operators", Editorial: Board (Institute of Physics Publishing Bristol and Philadelphia).

[4] C. Coello (2002) "Introducción a la Optimización Evolutiva multiobjetivo", http://www.cs.cinvestav.mx/~EVOCINV.

[5] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan (2002), "A fast and elitist multiobjective genetic algorithm: NSGA-II", IEEE transactions on evolutionary computation, vol. 6, Nº2:182-197.

[6] H. Kang, J. Park « A semi-fragile watermarking using JND »

[7] A. Nikolaidis and I. Pittas (2001) "Region-based image watermarking" IEEE Transactions on image processing, vol. 10, Nº11:1726-1740.

[8] P. B. Schnek (1999) "Persistent access control to prevent piracy of digital information" Proceedings of the IEEE, vol.87, Nº7:1239-1250.

[9] C. Vleeschouwer, J. F. Delaigle and B. Macq (2002) "Invisibility and application functionalities in perceptual watermarking – an overview" Proceedings of the IEEE, vol. 90, Nº1: 64-77.

[10] G.Voyatzis, I. Pitas (1999), "The use of watermarks in the protection of digital multimedia products", Proceedings of the IEEE, vol. 87, Nº7: 1197 – 1207.

[11] R. B. Wolfgang, C. I. Podilchuk, E. J. Delp (1999) "Perceptual watermarks for digital images and video", Proceedings of the IEEE, vol. 87, Nº7: 1108-1126

Fig 5. Watermark: Original and recovered from solution 2 after low-pass filtering with sigma = 50, 60, 70, 80 , 90, 100.