electronics

Article

A General Model for the Design of Efficient Sign-Coding Tools for Wavelet-Based Encoders

Otoniel Mario López-Granado [1,*], Miguel Onofre Martínez-Rach [1], Antonio Martí-Campoy [2], Marco Antonio Cruz-Chávez [3] and Manuel Pérez Malumbres [1]

[1] Department of Computer Engineering, Miguel Hernandez University, Avda. Universidad, s/n, 03202 Elche, Spain; mmrach@umh.es (M.O.M.-R.); mels@umh.es (M.P.M.)

[2] Instituto de Tecnologías de la Información y Comunicaciones ITACA, Universitat Politècnica de València, 46022 Valencia, Spain; amarti@disca.upv.es

[3] UAEM-Engineering and Applied Science Research Center, 62209 Cuernavaca, Mexico; mcruz@uaem.mx

* Correspondence: otoniel@umh.es

Received: 23 September 2020; Accepted: 10 November 2020; Published: 12 November 2020

Abstract: Traditionally, it has been assumed that the compression of the sign of wavelet coefficients is not worth the effort because they form a zero-mean process. However, several image encoders such as JPEG 2000 include sign-coding capabilities. In this paper, we analyze the convenience of including sign-coding techniques into wavelet-based image encoders and propose a methodology that allows the design of sign-prediction tools for whatever kind of wavelet-based encoder. The proposed methodology is based on the use of metaheuristic algorithms to find the best sign prediction with the most appropriate context distribution that maximizes the resulting sign-compression rate of a particular wavelet encoder. Following our proposal, we have designed and implemented a sign-coding module for the LTW wavelet encoder, to evaluate the benefits of the sign-coding tool provided by our proposed methodology. The experimental results show that sign compression can save up to 18.91% of bit-rate when enabling sign-coding capabilities. Also, we have observed two general behaviors when coding the sign of wavelet coefficients: (a) the best results are provided from moderate to high compression rates; and (b) the sign redundancy may be better exploited when working with high-textured images.

Keywords: iterated local search; sign coding; image coding; wavelet transform

1. Introduction

For a long time, wavelet transforms have been used in the scope of image compression and in fact many state-of-the-art image codecs employ a wavelet transform in their encoding engines [1–3], even the JPEG2000 standard [4] uses a wavelet transform stage. Wavelet transform provides both frequency and spatial localization of image energy. The energy of the image is gathered into a small fraction of the resulting transform coefficients. Although the energy of a wavelet transform coefficient is restricted to non-negative real numbers, the coefficients are defined by both a magnitude and a sign. In 1996, Shapiro stated in [2] that a wavelet transform coefficient has the same probability of being positive or negative and thus one bit should be used to encode the sign. However, several authors have proposed sign-coding strategies based on context modeling [3,5–9].

In Ref. [6], Deever and Hemami perform a deep analysis of sign coding in the scope of an embedded wavelet image coder. The paper shows that a PSNR improvement up to 0.7 dB is possible when sign entropy coding and a new extrapolation technique, based on the mutual information obtained from a biorthogonal basis vectors, are combined to improve the estimation of insignificant coefficients.

The Embedded Block Coding with Optimized Truncation (EBCOT) [5], included in the JPEG 2000 standard, encodes the sign of wavelet coefficients using context information which is obtained from the sign of horizontal and vertical neighbor coefficients (North, South, East, and West directions). EBCOT uses five contexts to model the sign-coding stage.

In Ref. [3], X. Wu presents a high order context modeling encoder. In this coder, the sign and the textures share the same context modeling. This model is based on a different neighborhood for the HL, LH and HH wavelet subbands. For the HL subband, the sign information of North, North-West, North-East, North-North and South sign is used to predict the current coefficient sign. The sign information used for the LH subband are from neighbors located at North, North-West, North-East, West-West and East directions. Finally, for the HH subband, an inter-band prediction is used besides the predictions used in the HL and LH subbands. So the neighborhood sign information used in the HH subband are North, North-West, North-East, C-HL and C-LH, where C-HL and C-LH are two sibling coefficients of the current coefficient that are at the same spatial location in the HL and LH subbands of the same wavelet decomposition level, respectively.

In Ref. [9], authors propose a method where wavelet coefficients sign is coded in a separate way. The sign and the magnitude of wavelet coefficients are examined to obtain their probabilities. The probabilities are obtained bit-plane by bit-plane and authors show that the sign information may be encoded by a probability of 0.5 only in the least significant five bit-planes. Great improvements are obtained when applying both sign-coding and refinement information.

All these works provide sign-coding methods that fully exploits the available sign information of neighboring wavelet coefficients. However, several authors have shown interest in developing very fast and simple wavelet encoders that are able to get reasonable good performance with reduced computing resource requirements [1,10–13]. Their design changed the way wavelet coefficients are encoded, following, most of them, the one-pass encoding scheme. That restricts the available sign information when encoding a single wavelet coefficient, being difficult to employ the sign-coding methods explained before. That is the reason these fast wavelet encoders do not use sign-coding tools. They encode each wavelet coefficient as soon as they are visited and use an additional bit to encode the wavelet coefficients sign.

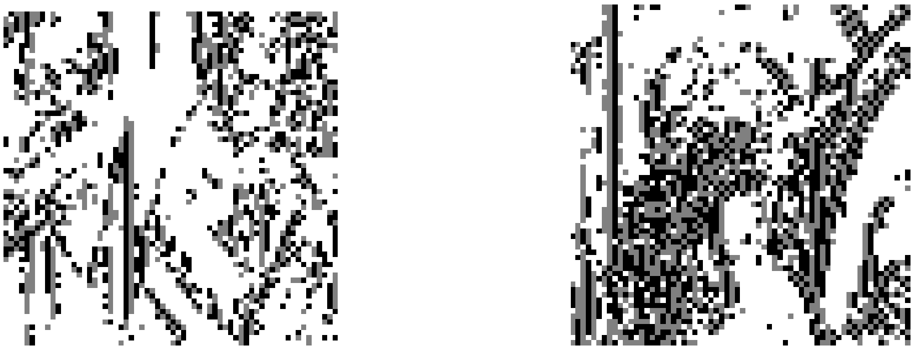
To better understand the potential sign correlations between wavelet coefficients, Schwartz, Zandi and Boliek [14] were the first authors, up to our knowledge, who propose a sign-prediction algorithm based on the sign information from neighbor wavelet coefficients, by using a context model. The main idea behind this approach is to find correlations along and across edges.

The HL subbands of a multi-scale 2-D wavelet decomposition are formed from low-pass vertical filtering and high-pass horizontal filtering. The high-pass filtering detects edges in the scanning direction, thus the HL subbands mainly contain vertical edge information as they result from high-pass filtering the lines of the input image. Oppositely defined are the LH subbands that contain primarily horizontal edge information.

As Deever explained in [15], given a vertical edge in an HL subband, it is expected that neighboring coefficients along that edge have the same sign as the coefficient being coded. This is mainly because usually, there is a high vertical correlation along vertical edges in images.

Moreover, we should consider correlation across edges, being the nature of the correlation directly related with the structure of the high-pass filter. For Daubechies' 9/7 filters, wavelet coefficient signs are strongly negatively correlated across edges because this filter is quite similar to a second derivative of a Gaussian as derived from theory of zero crossings and edge detection [16]. Therefore, it is reasonable to expect that wavelet coefficients will change sign as the edge is crossed. Although there is a sub-sampling process involved in the discrete wavelet transform, the sub-sampled coefficients remain strongly negatively correlated across edges. In this way, when a wavelet coefficient is optimally predicted as a function of its across-edge neighbors (e.g., left and right neighbors in HL subbands), the optimal prediction coefficients are negative, indicating an expected sign change. This conclusion is general for any wavelet with a shape similar to a second derivative of a Gaussian.

In Figure 1 we plot the spatial distributions of signs in the HL subband of two popular test images (Barbara and Lena), representing negative, positive and non-significant (null) wavelet coefficients with black, grey and white dots, respectively. The visible sign structures suggest that the sign bits of wavelet coefficients are compressible through proper context modeling techniques.


(a) Sign map for Barbara

(b) Sign map for Lena

Figure 1. Sign patterns in the HL subband for Barbara (**a**) and Lena (**b**) images. Black, grey and white dots represent negative, positive and non-significant coefficients, respectively [17].

2. Rationale

This work defines a methodology to find an efficient wavelet sign-coding tool for a particular wavelet-based encoder, does not matter if embedded or not. Once the wavelet-based encoder is chosen, it will determine which filter bank is used in the transform stage, the type of decomposition (dyadic, packet, etc.), number of decomposition levels, and the way in which resulting wavelet coefficients are encoded (i.e., scanning order, multipass/singlepass encoding, etc.). Therefore, by knowing the capabilities of the target wavelet encoder, our methodology is able to provide (a) an ad hoc sign-prediction table, and (b) a context formation that optimizes the entropy coding of the sign-prediction results.

Our methodology may be defined as a three stage process that it is applied only once to provide a sign-coding tool especially suited for the target wavelet-based encoder.

The first stage is devoted to build a sign-prediction table based on the sign information of the available wavelet coefficient neighbors. The target encoder will determine (a) the filter bank used in the wavelet transform, (b) the number of decomposition levels, and (c) the neighbor coefficients that will be available when encoding/decoding the sign of a wavelet coefficient. To predict the sign of a wavelet coefficient, we will use the sign of its neighbors and the existing correlation among them. The number of entries in the prediction table is the number of different combinations of neighbor sign values, as will be explained in the next section.

After building the prediction table, we can predict the sign value of a wavelet coefficient based on whatever combination of sign values of the predefined neighbors. The second stage of our methodology consist of distributing the prediction table entries in several contexts, in order to maximize the performance of the entropy coder that will encode the results of the sign predictions. This optimization process has a huge complexity, so we will need metaheuristic optimization approaches to find a good context distribution.

The last stage, will consist on combine the prediction table (from first stage) with the context distribution (from second stage) in a context prediction table that will be hard-wired at both encoder and decoder sides. Therefore, accessing to the prediction and its context will be very fast, a simple look-up table memory access.

In previous works, we presented a Genetic Algorithm (GA) [17,18] and a Simulated Annealing (SA) algorithm [19] to solve the context distribution of sign predictions. In [17] a deep analysis on how the population size and mutation probability parameters affect to the GA algorithm is performed.

In [19] a comparison between GA and SA algorithms is presented, but without employing context distribution of the sign prediction. The main innovations introduced in this paper are related with (a) a new methodology proposal to design and develop a simple sign-coding tool for whatever wavelet image codec, (b) a new Iterated Local Search (ILS) algorithm to solve the context distribution of sign predictions, comparing it with previous GA and SA algorithms in terms of convergence and goodness of the provided solution, and (c) a more detailed evaluation process of the sign-coding tool developed for the LTW wavelet encoder (extended test image set, and the use of Bjontegaard's Delta Rate (BD-rate) R/D metric [20]).

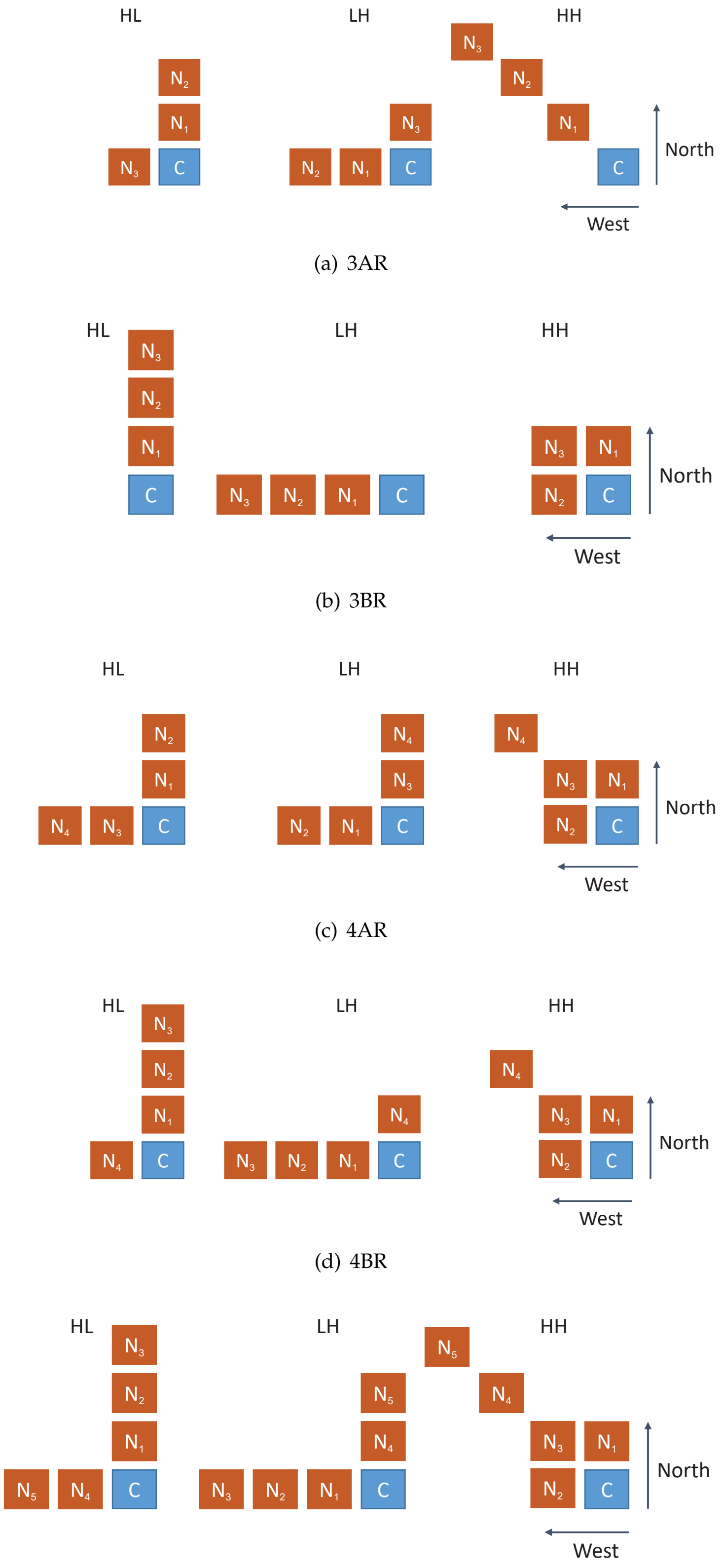
The rest of the paper is organized as follows: The next section describes how prediction table is built. The optimal context distribution section describes the metaheuristic algorithm used to find out the best context distribution of the sign-prediction table entries. The evaluation of the proposed algorithm as well as a comparison against other state-of-the-art algorithms is also presented. In the sign-coding evaluation section we will apply our methodology to design a sign-coding tool especially suited for a particular wavelet encoder, LTW, in order to evaluate sign-coding performance in a real implementation. Finally, some conclusions are presented.

3. Building the Sign-Prediction Table

In this section, we will explain how the sign-prediction table should be built, analyzing the sign redundancy among wavelet coefficients belonging to the same wavelet decomposition subband type (HL, LH, and HH). The sign redundancy will strongly depend on the wavelet filter bank employed in the dyadic decomposition. This redundancy may be captured taking the contextual sign information from the coefficients belonging to the neighborhood of the current coefficient from whom its sign will be predicted. Therefore, the definition of the neighborhood should be done with care to properly capture the sign redundancy at each subband type. To estimate sign correlation in a practical way, we have applied a Daubechies 9/7F wavelet filter bank (most popular filter bank) with a 6-level dyadic decomposition to each image from a representative set of natural images (i.e., Kodak set [Kodak]). If the encoder uses a different filter bank or a different number of decomposition levels, the procedure will be the same.

As Deever explained in [6], the sign neighborhood correlation depends on the subband type (HL,LH,HH), so, we have used different neighbors in each subband type, trying to exploit the correlation found around edges. Therefore, for a particular subband type, we have defined n neighbors that can hold one of the three possible sign values that are positive, negative and null (zero), labeled as "$+$", "$-$" and "$*$" respectively. This lead us to a set of 3^n different Neighbor Sign Patterns (NSP) for each subband type, n being the number of neighbors used in the sign prediction of a particular wavelet coefficient. For example, Table 1 shows the 3^2 NSPs for a neighborhood composed only by two neighbors.

In Figure 2, different neighborhoods are shown for a particular wavelet-based encoder. Each neighborhood is composed by several neighbors (3, 4 or 5). To specify the neighbors position, relative to the current coefficient, we use cardinal direction steps to reach them. So, neighborhood 3AR in the HL subband is composed by three neighbors located, one position in north direction (N), two positions in north direction (NN) and one position in west direction (W). The cardinal directions used are different depending on the subband type (HL, LH and HH). When two neighborhoods hold the same number of neighbors we differentiate them using the letter A of B. The letter R in the name of the neighborhood stands for raster order. In this case our target wavelet-based encoder has a restriction on the available neighbors: only those wavelet coefficients yet encoded (in raster order) are available to form a neighborhood. Notice that neighborhoods could be defined with more than three cardinal directions, depending on the availability of neighbors sign information at encoding/decoding stage. This is the case of other wavelet encoders such as JPEG2000 and the one proposed by [8] which use the four cardinal directions (N,S,E,W) for the context formation.



(a) 3AR

(b) 3BR

(c) 4AR

(d) 4BR

(e) 5AR

Figure 2. Set of potential Neighborhoods for a particular wavelet-based codec. C is the current wavelet coefficient and N_i are its neighbors, where 3AR (**a**) and 3BR (**b**) use three neighbors, 4AR (**c**) and 4BR (**d**) use four neighbors, and 5AR (**e**) uses five neighbors for the sign prediction of C wavelet coefficient.

In Table 2 we show the probability distribution of the Neighbor Sign Pattern (NSP) in the HL_6 subband (at the sixth wavelet decomposition level) of the Lena test image when using the neighborhood 3AR. As shown, the probability that a wavelet coefficient C (first column) is positive when its N, NN and W neighbors are also positive (first NSP in Table 2) is 20.31%. Moreover, when the N and NN neighbors have the same sign that current coefficient C and the W neighbor has the opposite sign (rows 2 and 3), then the current coefficient C has the same sign than its northern neighbors with a probability of 25%.

Table 1. The 9 Neighbor Sign Patterns (NSPs) for a 2-neighbors neighborhood.

NSP	First Neighbor	Second Neighbor
1	+	+
2	+	−
3	+	*
4	−	+
5	−	−
6	−	*
7	*	+
8	*	−
9	*	*

Table 2. Probability distribution of neighbor sign patterns (NSPs) of HL_6 subband (8 × 8 coefficients) in Lena image. In frequency column we show the total occurrences of the corresponding NSP.

C	NSPs			Frequency	%Probability
	N	NN	W		
+	+	+	+	13	20.31
+	+	+	−	8	12.50
−	−	−	+	8	12.50
−	+	+	+	6	9.38
−	−	+	+	6	9.38
...	... (others)			23	35.93

Therefore, after analyzing all the images from the representative image set, we obtain the sign prediction table for each subband type and its particular neighborhood set-up. The prediction table will have just one entry for each $NSP\,[k]\,, k = 1, ..., 3^n$. Then, when coding the sign of a wavelet coefficient in a particular subband, first we will get the sign value of its neighborhood to form the current NSP. Then, we will look up the NSP in the prediction table to find the sign prediction of the current wavelet coefficient. Therefore, at the end we have a prediction, and therefore what we are going to encode is the correctness of that prediction, i.e., a binary valued symbol (1: success; 0: failure) resulting from Equation (1), where $SC_{i,j}$ is the sign of the coefficient located at position i, j and $\hat{SC}_{i,j}\,[k]$ is the sign prediction based on the neighborhood $NSP\,[k]$. $SC_{i,j}$ and $\hat{SC}_{i,j}\,[k]$ values can be 0 (positive) or 1 (negative), because the sign prediction is only performed over significant wavelet coefficients after quantization. Consequently, the *PredictedSymbol* will be 1 (success) if the values of both current wavelet coefficient sign and its sign prediction are the same.

$$PredictedSymbol = \overline{\left(\hat{SC}_{i,j}\,[k] \oplus SC_{i,j}\right)} \tag{1}$$

Therefore, the performance of our binary entropy encoder will depend on the behavior of our sign predictor, the higher success prediction, the higher compression is achieved. In Table 3, we show the resulting prediction tables for each subband type when using the 3AR neighborhood. In this table, for each subband, N_1, N_2, and N_3 are the corresponding neighbor's sign values, which can be + (positive), − (negative), or * (NULL).

To obtain a higher compression performance of the entropy encoder, we propose the use of r contexts. In this manner, for each subband type, we distribute the provided NSPs sign predictions into

r sets (contexts) in such a way that the overall aggregate entropy will be reduced and as a consequence higher compression rates can be achieved. The task of distributing the NSPs sign predictions into several contexts to maximize the coding performance of sign prediction is a hard optimization problem that will be addressed in the next section.

Table 3. Sign-prediction table of Barbara image for HL, LH and HH subbands. For each subband type, N_1, N_2 and N_3 are the corresponding neighbor's sign values.

k		**NSP**		**HL Subband (N,NN,W) Prediction**	**LH Subband (W,WW,N) Prediction**	**HH Subband (N,W,NW) Prediction**
	N_1	N_2	N_3	$(\hat{SC}_{i,j}\,[k]_{HL})$	$(\hat{SC}_{i,j}\,[k]_{LH})$	$(\hat{SC}_{i,j}\,[k]_{HH})$
0	*	*	*	+	+	+
1	*	*	+	−	−	+
2	*	*	−	+	+	−
3	*	+	*	+	+	−
4	*	+	+	−	−	−
5	*	+	−	+	+	−
6	*	−	*	−	−	+
7	*	−	+	−	−	+
8	*	−	−	+	+	+
9	+	*	*	+	−	−
10	+	*	+	−	−	−
11	+	*	−	+	+	−
12	+	+	*	+	+	−
13	+	+	+	−	−	−
14	+	+	−	+	+	−
15	+	−	*	−	−	+
16	+	−	+	−	−	−
17	+	−	−	+	+	+
18	−	*	*	−	+	+
19	−	*	+	−	−	+
20	−	*	−	+	+	+
21	−	+	*	+	+	+
22	−	+	+	−	−	−
23	−	+	−	+	+	+
24	−	−	*	−	−	+
25	−	−	+	−	−	+
26	−	−	−	+	+	+

4. Determining the Optimal Context Distribution

Once the prediction table is obtained for each neighborhood, the next step is focused on finding a context distribution of its NSPs for each subband type SB_t, where $t = \{HL, LH, HH\}$ that maximizes the sign-compression rate.

For the sake of simplicity, we will use the 3AR neighborhood. Therefore, as shown in Table 3, for a particular wavelet coefficient $C_{i,j}$ with sign value $SC_{i,j} = \{+, -\}$ that belongs to subband SB_t, the prediction table will provide a sign prediction, $\hat{SC}_{i,j}$, based on its Neighborhood Sign Pattern, $NSP\,[k]$, where k is the index of the NSP that matches with the $C_{i,j}$ neighborhood sign values.

There is no an univocal relationship between a NSP and the sign prediction in a subband type, i.e., not always for a particular $NSP\,[k]$, the sign prediction $\hat{SC}_{i,j}$ is positive or negative. However, it is possible to find out that for a particular $NSP\,[k]$, the predicted $\hat{SC}_{i,j}$ is more probable to be positive than negative or vice versa. However, the problem is still more complex, because a sign prediction for a NSP could fit well for an image but not for others. Even more, the context distribution and the number of contexts to use will affect the encoder compression performance. Therefore, the idea is to find a context distribution that better fits for a representative set of images, so we can capture the canonical wavelet sign redundancy introduced by a particular wavelet filter. In this manner, once the universal context distribution of the sign-prediction table is found, it could be hard-wired at both encoder and decoder sides.

At this point, the motivation of using optimization algorithms to compress the sign of wavelet coefficients is two-fold. First, when the number of selected neighbors for sign correlation analysis grows, or when the number of contexts to distribute the NSPs in grows, the search space is excessively wide. Second, it is not intuitive to find an efficient strategy to group in contexts different NSPs to maximize the entropy sign-coding performance.

In fact, the context distribution problem is similar to the problem of finding the partitions of a set of k objects into r subsets (clustering data sets). The number of possible partitions is called Stirling number of the second kind [21] and is denoted by $S(k,r)$ (see Equation (2)). For example, if we use three neighbors for the sign prediction, we have 27 NSPs (3^3) and if we distribute then into 5 contexts, the Stirling number of the second kind is $S(27,5) = 61,338,207,158,409,090$, which means that we have that number of possibilities to distribute the 27 NSPs into 5 contexts. This huge number shows the complexity and magnitude of the problem, even more when we increase both the number of neighbors and the number of contexts.

$$S(k,r) = \frac{1}{r!} \sum_{j=0}^{r} (-1)^{r-j} \binom{r}{j} j^k \tag{2}$$

To solve the context distribution problem addressed above, we will evaluate different optimization algorithms to find the one that better fit to the context distribution optimization problem. In following subsection, we will introduce the iterated local search optimization algorithm, showing how the algorithm is adapted to this optimization problem.

4.1. Iterated Local Search

The Iterated Local Search (ILS) algorithm [22] is a global-local search optimization method that does not focus the search on the full space of candidate solutions, but on the solutions that are provided by some underlying algorithm, typically a local search heuristic. Local search [23] is based on which perhaps is the oldest optimization method ′trial and error′. Local search algorithms start with an initial solution, which is altered by means of operators that modify the solution. Then, if the altered solution is better than the original one, it is replaced, in other case, it returns to the initial one. The procedure is repeated until no improvement is achieved in the solution. These procedures are usually called hill climbing. However, the local search algorithm only finds the local maximum that it is closer to the starting point. To avoid this, the iterated local search after locating a local maximum will jump to other point of the solution space to start a new local search procedure, in order to find better solutions.

The quality of the local optima obtained by a local search method depends on the initial solution. As we can generate local optima with high variability, iterated local search may be used to improve the quality of successive local optima [24]. The general local search algorithm [23] starts with an initial feasible solution $s \in S$, and use a function that looks for a better solution within its own neighborhood. So long as an improved solution exists, it is adopted and the neighborhood search is repeated starting from the new solution. When a local optimum is reached, the local search stops. ILS [22] achieves an intermediate state $s' \in S$, applying a change or perturbation to a given current solution s. If the perturbation applied is too high, the search process becomes global in the search space, but if the perturbation applied is low, the search process becomes local to a solution neighborhood. After applying the perturbation, local search is applied to s' reaching a new solution $s^{*'} \in S^*$. If $s^{*'}$ meets the acceptance criteria, it becomes the next element in S^*; otherwise, it returns to s^*. The resulting path is the outcome of applying stochastic search in S^*, but where neighborhoods are never explicitly introduced. The ILS should lead to a good-biased sampling if the perturbations neither are too small nor too large.

Before starting with the design of the ILS algorithm, we need to obtain the sign-prediction table for each wavelet subband type, as explained in the previous section. Once the sign-prediction tables are available, we proceed to find a context distribution for each subband type by grouping the NSPs found at sign-prediction table in independent sets (or contexts). As we need a context distribution for each subband type, we will only focus on one of them, taking into account that the context distribution of the other subband types will be obtained running the same algorithm with their own neighborhood configurations.

The way to map our problem to the definitions and processes required by a ILS algorithm, described above, is to create an initial solution that will improve their goodness (cost) during the iteration process. The goodness of each solution is defined by a cost function that will determine their quality (see Equation (3)). Every individual (a particular context distribution) is defined with: the sign-prediction table, the associated context of each $NSP\,[k]$ and its current cost value. For our purposes, we will define a cost function that measures the sign-compression performance of each individual (bit savings (% gain)). In other words, the cost function will score the compression rate that would be achieved if we use the context distribution defined by this individual.

$$Cost = 100 - \left(\frac{B_t}{O_t} * 100\right) \tag{3}$$

where B_t corresponds with the total bits resulting of the compression of all wavelet coefficient signs using the NSP context distribution provided by the proposed algorithm (see Equation (4)) and O_t represents the total number of significant (non-null) wavelet coefficients found in the analyzed image set. As can be seen in Equation (4), B_t is calculated as the sum of an estimation of the bits required to compress the sign of the wavelet coefficients belonging to each defined context ($r_i, i = 1, ..., N$). This estimation is based on (a) the entropy of context r_i ($H_{r_i}(X)$ see Equation (5)), and (b) the sum of the number of those wavelet coefficients that have an NSP belonging to context r_i (notice that context r_i have $|r_i|$ different NSPs).

$$B_t = \sum_{i=1}^{N}\left(H_{r_i}(X) * (\sum_{n=1}^{|r_i|} O_n)\right) \tag{4}$$

$$H_{r_i}(X) = \sum_{n=1}^{|r_i|}\left(-\sum_{i} P_n(x_i) * log_2(P_n(x_i))\right) \tag{5}$$

Equation (5) shows the entropy of a given context r_i, where X is a discrete random variable with two possible values (x_i), success or failure in the sign prediction of a given wavelet coefficient. P_n is the probability of success in the sign prediction of the n_{th} NSP that belongs to context r_i.

Once the cost function is defined, in Algorithm 1 we present the ILS pseudo code that will find the optimal/suboptimal context distribution of wavelet sign predictions. First of all, an initial solution s is defined by (a) a random context distribution of the NSPs of the prediction table, and (b) its corresponding cost obtained from the objective function (Equation (3)). A local search is then invoked to obtain a local optimum solution (s^*).

Algorithm 1: Iterated Local Search algorithm for context distribution of sign predictions.

```
Input: Context_Number, PredictionTable, $N$
function IteratedLocalSearch
    $s$ = GenerateInitialSolution()
    EvaluateCostFunction(s)
    $s^*$ = LocalSearch(s)
    repeat
        $s'$ = PerturbationSolution($s^*$)
        EvaluateCostFunction($s'$)
        $s^{*'}$ = LocalSearch($s'$)
        if ($f(s^{*'}) \geq f(s^*)$)
            $s^* = s^{*'}$
        end
    until $N$ iterations
end of function
```

As stated before, a good perturbation transforms one solution, s^*, into an excellent starting point, s', for a new local search. The ILS algorithm will replace s' for a new solution $s^{*'}$, as long as the cost of the objective function $f(s^{*'})$ is better than $f(s^*)$. After N iterations, the ILS algorithm ends, with an improved context distribution for a given *PredictionTable* associated to a subband type. However, there is no guarantee that this distribution is the global optimum for this problem.

An exhaustive tuning of the ILS parameters has been done through a sensitivity analysis based on [25], in order to allow ILS algorithm to find out a solution with the best quality as possible. Stopping criteria for local search algorithm was set as twice the size of the NSPs that depends on the neighborhood configuration, $(3^n * 2)$ for local search loop. The number of iterations of the ILS algorithm (N) that best results obtained was 10. After the analysis of several ILS perturbation schemes, we have chosen a perturbation schema in which eight NSPs of the best solution, reported by the local search, are randomly changed. The position of the first NSP to be changed was randomly selected, and the rest of them are evenly distributed. After running the ILS for each wavelet subband type, a solution is obtained containing the context distribution of sign predictions for a given neighborhood.

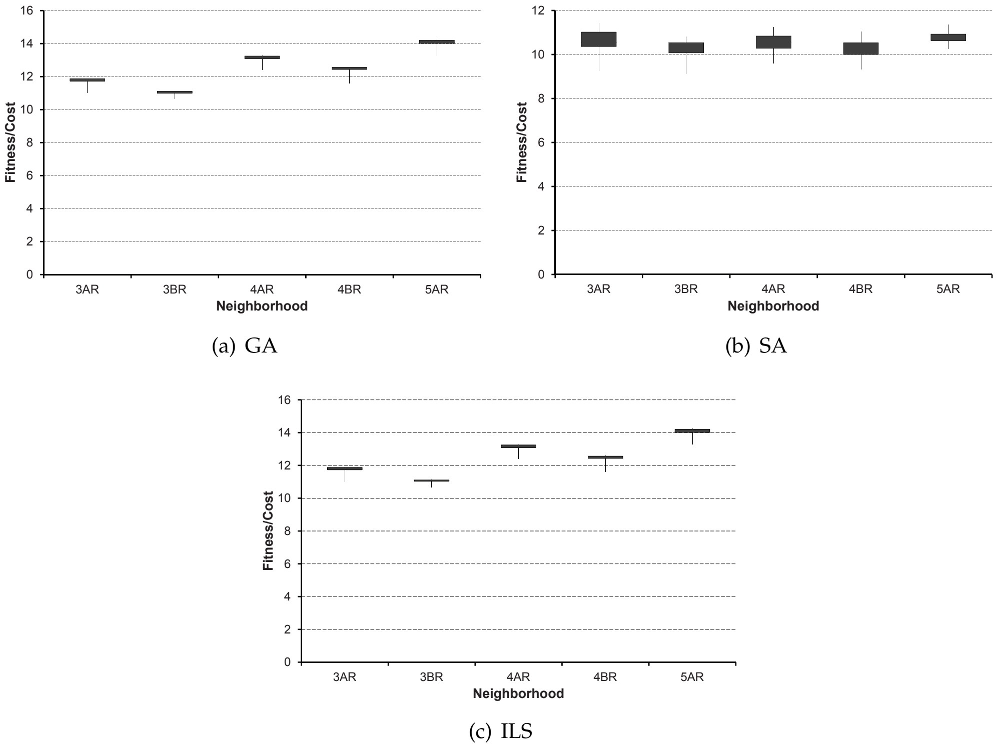
4.2. Evaluation of the Proposed ILS Algorithm

In this section, we analyze the behavior of the proposed algorithm in terms of convergence, complexity and goodness of the resulting solution. Our purpose is to determine if the ILS algorithm fits to the context distribution optimization problem. Also, we will find the number of contexts and neighborhood set-up that provide the best solution (maximum estimated bit-rate savings). Once the best sign-prediction table and context distribution are computed, we can include them in the sign-coding/decoding tool.

In previous works [17–19] authors presented a Genetic Algorithm (GA) and a Simulated Annealing (SA) algorithm to solve the context distribution of sign predictions problem. In this paper, we will compare previous proposals with the new proposed ILS algorithm in terms of convergence and goodness of the resulting solution.

We have tested all the algorithms for different neighborhoods (see Figure 2) using the set of representative images conformed by Lena (512 × 512), Barbara (512 × 512) and the whole 23 images of the Kodak set [26] (768 × 512). The platform used to perform the evaluation is an Intel Pentium Dual Core 3.0 GHz with 4 Gbyte RAM memory. To avoid the effect of the random seed in the output of the algorithms, the algorithms have been executed several times for each scenario. In our experiments, we run 30 times each algorithm for each combination of image set, neighborhood and number of contexts. From all these runs we can evaluate the goodness of the algorithms with consistent statistical results that will allow us to analyze the quality of the provided solutions.

Figure 3 shows the fitness/cost dispersion for all algorithms as a function of the neighborhood for HL subband. As can be seen, GA and ILS algorithms have the same behavior with better solutions and lower dispersion in all executions than SA algorithm for all evaluated neighborhoods. Furthermore, SA algorithm ranks the goodness of the neighborhoods in a different way than GA and ILS algorithms. For GA and ILS algorithms, neighborhood 5AR shows the best fitness value. However, although there are slight differences, for the SA algorithm, the best fitness value is obtained using the 3AR neighborhood configuration. As all neighborhood configurations provide similar results, the SA results break the rule that as more neighbors used, more correlation is found, and better compression is achieved.



(a) GA

(b) SA

(c) ILS

Figure 3. Fitness/cost dispersion vs. Neighborhood when applied to HL subband. (**a**) For GA algorithm. (**b**) For SA algorithm. (**c**) For ILS algorithm.

Now, we will evaluate the fitness/cost dispersion regarding the number of contexts focusing on the 5AR neighborhood and HL subband. As shown in Figure 4 as the number of contexts increases the fitness value does. All algorithms show this effect, but SA algorithm shows a different trend and a greater dispersion than the GA and ILS algorithms.

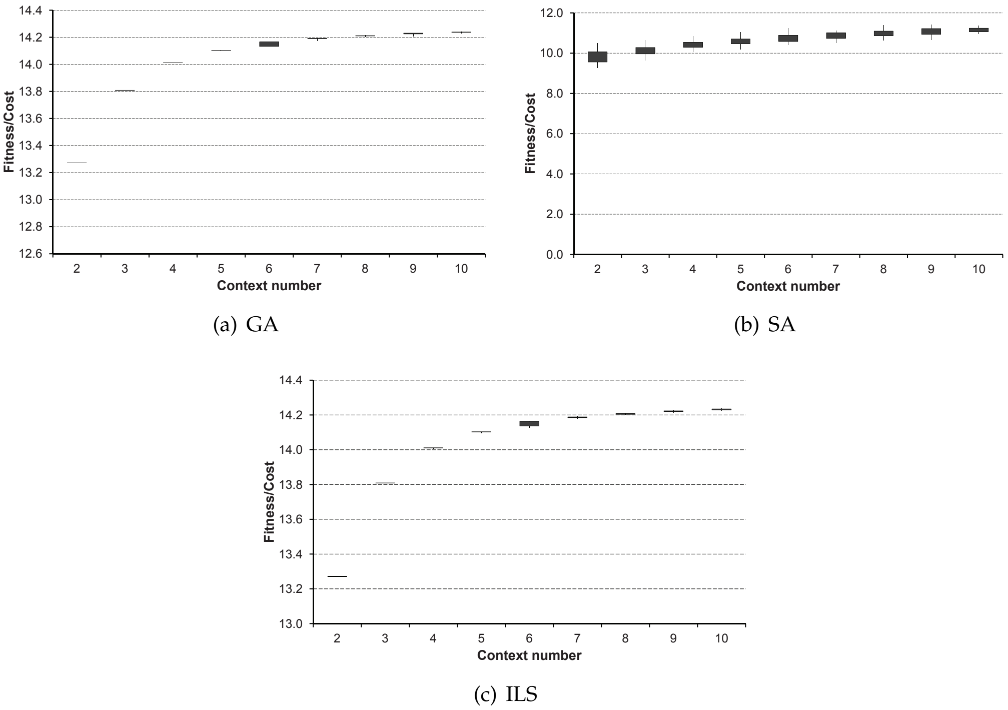
Therefore, looking at the presented results, we could assess that both GA and ILS performs better than SA algorithm, because they obtain better fitness values with a lower dispersion in all the executions. Although not shown, the behavior is similar when applied to LH or HH subbands. We have selected the ILS algorithm because it has the lowest computational complexity mainly since it uses a population size of 1 and no recombination operators are needed [22]. In our experiments, ILS algorithm is up to 25% faster than GA algorithm (13% on average), providing solutions of similar quality/goodness.

From the analysis of the evaluation results, in Table 4, we propose the parameters that provide the best solution for an image wavelet encoder that uses (a) the 9-7 biorthogonal filter bank and (b) the sign information from previous neighbors in raster order.

Table 4. Proposed parameters to get the best solution.

Neighborhood Configuration	**5AR**
Context number	10
Initial population	Randomly created
Stopping criteria for local search	$3^5 \times 2$ (twice the size of NSP)
Perturbation schema	Change of eight NSPs, the first one randomly chosen and evenly distributed the remaining seven
Number of iterations	10

To apply to wavelet encoders with different requirements than the ones used in this work, the whole methodology will be applied following the same steps to obtain the best solution that fit with the features of these wavelet image encoders.



(a) GA

(b) SA

(c) ILS

Figure 4. Fitness/cost dispersion vs. Context number applied to HL subband for neighborhood 5AR. (**a**) For GA algorithm. (**b**) For SA algorithm. (**c**) For ILS algorithm.

5. Sign-Coding Evaluation

In this section, we will use the solution provided by the ILS algorithm with the proposed configuration to implement a sign-coding tool in a real encoder. We will show its real (not estimated) impact on the encoder performance. We have chosen the LTW wavelet encoder [11] as it has the same requirements (filterbank, neighborhood, etc.) that the ones followed in our evaluation study in the previous section. The target is to design a new sign-coding module based on the solution reported by our framework that will be implemented in the LTW image wavelet encoder. The new resulting encoder is called S-LTW (Both LTW and S-LTW binaries as well as ILS source code will be available under request).

We performed several experimental tests comparing S-LTW encoder with LTW, JPEG2000 (Jasper 1.701.0) and SPIHT (Spiht 8.01) in terms of R/D. All the evaluated encoders have been tested on an Intel Pentium Dual Core 3.0 GHz with 4 Gbyte RAM memory. The correspondent encoder binaries were obtained by means of Microsoft Visual C++ (2008 version) compiler with the same project options. The test images used in the evaluation were: Bike (2560 × 2048), GoldHill (512 × 512), Cafe (2560 × 2048), Peppers (512 × 512), Zelda (512 × 512), Woman (2560 × 2048) and a set of higher resolution images from [27].

5.1. Compression Performance of the Sign-Coding Proposal

First, we are going to determine the compression performance of the proposed sign-coding scheme alone. As no sign-coding technique is used, the sign information is raw encoded, so it would require so many bits as the number of significant coefficients (one bit per significant coefficient). In order to obtain the relative compression gains of the proposed sign-coding scheme, for each target bit-rate, we have counted the total number of significant coefficients to be encoded just after quantization.

For example, the LTW encoder at 1 bpp must encode 855,266 significant coefficients for Bike test image, requiring 855,266 bits for coding the sign information. If we include the proposed sign encoding tool, the encoded sign information will be reduced to 740,066 bits, representing a 13.47% of bit-rate savings for the sign information.

In Table 5 we show the relative compression gains with respect to the original encoders due only to the sign-coding capability for all test images. As we can see, the maximum sign-compression gain is 18.91% for NightShot-ISO-100 image at 0.25 bpp. The compression gain is higher at moderate compression rates. However, as the compression rate increases, the number of non-significant coefficients increase, losing prediction performance since the NSPs with non-significant neighbors would be more and more dominant, losing the sign correlation info among neighbors. This effect results in reducing the compression gains at 0.125 bpp rates in all tested images, being especially noticed with the low resolution images (512 × 512). Furthermore, the compression gain is greater for high-textured images such as Bike or Artificial as these kind of images better exposes the sign correlation of the wavelet filter bank.

Table 5. Sign-compression performance at different bit rates for S-LTW encoder.

Image	**%Gain**			
	1 bpp	**0.5 bpp**	**0.25 bpp**	**0.125 bpp**
Goldhill (512 × 512)	4.83	4.41	4.11	3.88
Peppers (512 × 512)	3.30	7.60	7.46	6.95
Zelda (512 × 512)	4.28	8.35	7.85	5.30
FlowerFoveon (1920 × 1280)	3.42	5.43	7.56	7.74
Cathedral (1920 × 2560)	7.35	8.90	8.47	6.93
Fireworks (2560 × 1920)	7.41	12.91	12.51	12.13
Hdr (2560 × 1920)	6.34	9.60	9.95	8.92
Leaves-ISO-200 (2560 × 1920)	10.46	8.59	6.42	4.80
NightShot-ISO-100 (2048 × 2560)	10.68	16.90	**18.91**	17.56
Bike (2048 × 2560)	13.47	15.63	15.32	13.07
Woman (2048 × 2560)	10.25	9.81	8.39	6.93
Cafe (2048 × 2560)	9.26	6.93	4.69	3.92
Artificial (3072 × 2048)	14.61	15.39	13.80	12.10
Bridge (2560 × 3840)	9.12	10.65	14.05	15.50
Deer (3840 × 2560)	5.25	3.59	4.00	4.63
SpiderWeb (3840 × 2560)	8.18	6.02	8.70	13.00
BigTree (5760 × 4480)	5.25	6.61	10.10	10.58
BigBuilding (7040 × 5120)	12.08	12.68	10.65	8.26

5.2. Impact of Sign-Coding Proposal on the Overall R/D Performance

To evaluate the impact in R/D of the proposed sign-coding scheme, we will compare the LTW encoder with and without the proposed sign-coding scheme against SPIHT and JPEG2000 encoders, so we will determine the real improvements achieved by including sign-coding capabilities to the LTW wavelet encoder. In Table 6, we show the Bjontegaard's Delta Rate (BD-rate) [20] for all tested images between S-LTW and JPEG2000 and between S-LTW and SPIHT. Positive values in Table 6 indicates that LTW or S-LTW obtains better results than JPEG2000 or SPIHT. When compared to LTW we can see that JPEG2000 and SPIHT have a better performance with respect to LTW in 5 and 9 images of the 18 tested images, respectively (no sign-coding). By introducing the sign-coding tool, the S-LTW encoder improves the original LTW codec, losing performance only in 2 of the 18 images with respect to JPEG2000 and SPIHT. As can be seen, the inclusion of the sign-coding capabilities has a real benefit in the R/D performance of the codec. The results show that on average, the S-LTW encoder achieves BD-rate improvements of 4.7% and 2.3% with respect to JPEG2000 and SPHIT, respectively. Therefore, by adding the sign-coding tool to the LTW wavelet encoder, its performance, in terms of BD-rate gains, increases around 170% (1.7×) on average, showing the real impact of sign coding in the overall encoder performance.

Table 6. %BD-Rate comparison between LTW and S-LTW respect to JPEG2000 and SPIHT for all tested images.

Image	**LTW**		**S-LTW**	
	JPEG2000	**SPIHT**	**JPEG2000**	**SPIHT**
Goldhill (512 × 512)	0.5%	−0.6%	1.4%	0.1%
Peppers (512 × 512)	3.9%	−0.7%	5.5%	0.4%
Zelda (512 × 512)	1.0%	−0.4%	2.7%	1.2%
FlowerFoveon (1920 × 1280)	16.6%	−2.6%	18.7%	−1.2%
Cathedral (1920 × 2560)	−1.2%	−0.4%	0.3%	1.0%
Fireworks (2560 × 1920)	4.3%	−3.3%	6.5%	−1.4%
Hdr (2560 × 1920)	2.6%	−1.8%	4.2%	0.1%
Leaves-ISO-200 (2560 × 1920)	1.8%	0.8%	3.1%	2.2%
NightShot-ISO-100 (2048 × 2560)	0.1%	−0.2%	3.8%	2.8%
Bike (2048 × 2560)	−5.8%	2.7%	−2.9%	5.3%
Woman (2048 × 2560)	−0.2%	1.1%	1.6%	3.0%
Cafe (2048 × 2560)	−1.8%	1.9%	−0.5%	3.1%
Artificial (3072 × 2048)	2.7%	−0.3%	5.3%	2.0%
Bridge (2560 × 3840)	−2.0%	1.0%	0.2%	3.1%
Deer (3840 × 2560)	1.1%	1.5%	2.2%	2.4%
SpiderWeb (3840 × 2560)	20.8%	1.9%	21.2%	3.2%
BigTree (5760 × 4480)	3.8%	2.3%	5.2%	3.7%
BigBuilding (7040 × 5120)	3.5%	7.2%	5.8%	9.7%
Average	**2.9%**	**0.6%**	**4.7%**	**2.3%**

In Figures 5 and 6 we graphically show the R/D improvement when comparing original LTW versus JPEG2000/SPIHT and S-LTW versus JPEG2000/SPIHT for Bike and Woman images, respectively. As shown, there is an increase in the PSNR difference ($\Delta PSNR$) between SPIHT and the LTW encoder when we enable the sign-coding capability. Also, regarding JPEG2000, we can see that now S-LTW has a minor loss in PSNR than original LTW for Bike image, keeping a difference of less than 0.2 dB in most of the bit-rate range. However, for Woman image, we can see a PSNR gain of up to 0.16 dB.

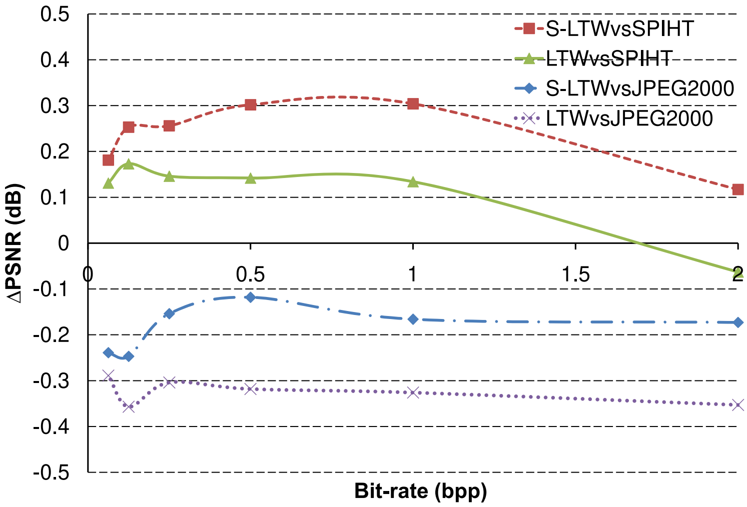


Figure 5. PSNR Gain for Bike.

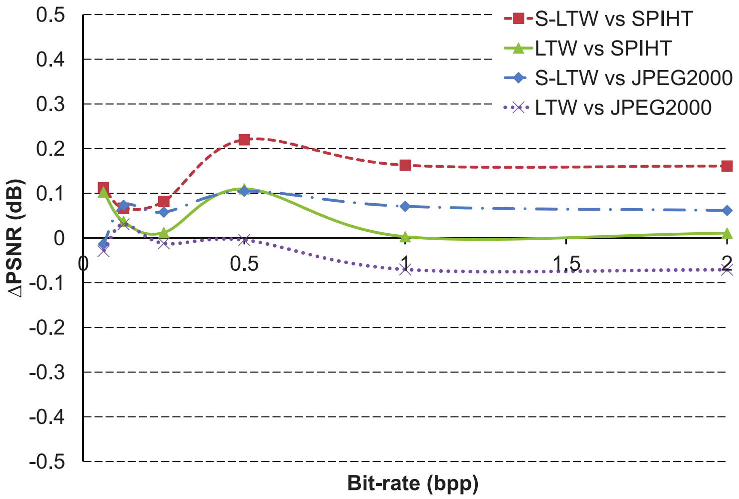


Figure 6. PSNR Gain for Woman.

Regarding to the encoder complexity, we want to remark that there is no significant increase in the encoder because once we have performed the prediction using ILS algorithm, the solution provided by the algorithm is hard-wired in both encoder and decoder and thus, the only extra work performed by the codec is the access to the look-up table to find the context for a given neighborhood.

The resulting methodology when applied to the LTW encoder has improved its R/D performance up to 0.37 dB. This result is in consonance with the ones obtained in [6] where improvements of up to 0.43 dB of reconstructed image quality were obtained when applying sign-coding techniques in their image codec. We also foresee similar R/D improvements in other wavelet codecs such as BCWT [13], as their coding architecture is very similar to the one exhibited by the LTW codec (same wavelet filter bank, same one-pass coding order and same available neighbors for sign prediction).

However, in order to see the benefits of our methodology in any other encoder, a similar process than the one presented for the LTW encoder must be followed. First, we must evaluate the internal characteristics of the target image encoder (i.e., wavelet filter bank, scanning order to see the available neighborhood, bit plate coding or not, etc.). Then, we must run the ILS algorithm with the determined neighborhood and the desired context number for the arithmetic encoder to obtain the specific sign predictor. Once we have defined the sign predictor, it should be integrated into the image coding architecture.

6. Conclusions

We have presented a deep study about sign coding for wavelet-based image encoders. We propose a methodology to design wavelet sign-coding tools that can be applied to any kind of wavelet-based image/video encoders. This methodology is based on metaheuristic algorithms to maximize the successful prediction of the sign for every significant wavelet coefficient. From the evaluation process, we have determined that the ILS optimization algorithm is able to provide the best solution to the wavelet coefficient sign-coding module, being slightly faster than the previous state-of-the-art algorithms.

To determine the benefits of the wavelet sign predictor provided by our methodology, we first need to choose a target wavelet codec (i.e., LTW) to extract its main features. Then, as explained in Sections 3 and 4, the proposed methodology is applied to provide a wavelet sign predictor specifically designed for the LTW codec. After implementing the corresponding sign-prediction module in the LTW codec, we have performed and R/D performance evaluation by measuring the performance of the sign-coding tool alone and also its impact on the overall R/D performance. The results show that by including the sign-coding tool, the reconstructed image quality increases up to 0.37 dB, being greater the improvement at moderate rates, especially when working with high-textured images. Furthermore, we have evaluated the S-LTW BD-rate performance to determine the impact of the sign-coding tool on the global encoding performance. The obtained results show that the LTW BD-rate is boosted 1.7 times, in average, when enabling sign-coding tool, increasing its performance when compared with JPEG2000 and SPIHT.

Finally, after the evaluation, we have confirmed the benefits of using our methodology to design a wavelet sign-coding tool for a particular wavelet encoder, showing that the sign-prediction capabilities it provides are able to improve the overall wavelet coding R/D performance.

As future work, we plan to perform an exhaustive study about training data quality that could help us to increase the prediction performance by selecting the appropriate image set, i.e., those images that better capture the wavelet sign correlation.

Author Contributions: Conceptualization, M.P.M. and O.M.L.-G.; methodology, O.M.L.-G. and M.P.M.; software, O.M.L.-G., M.O.M.-R. and A.M.-C. and M.A.C.-C.; validation, A.M.-C., O.M.L.-G. and M.P.M.; formal analysis, M.A.C.-C. and M.P.M.; writing–original draft preparation, A.M.-C., M.O.M.-R. and O.M.L.-G.; writing–review and editing, O.M.L.-G., A.M.-C. and M.P.M.; funding acquisition, O.M.L.-G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Spanish Ministry of Economy and Competitiveness under Grant RTI2018-098156-B-C54, co-financed by FEDER funds (MINECO/FEDER/UE).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Said, A.; Pearlman, A. A new, fast and efficient image codec based on set partitioning in hierarchical trees. *IEEE Trans. Circuits Syst. Video Technol.* **1996**, *6*, 243–250. [CrossRef]
2. Shapiro, J. A fast technique for identifying zerotrees in the EZW algorithm. In Proceedings of the 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference, Atlanta, GA, USA, 9 May 1996; pp. 1455–1458.
3. Wu, X. High-order context modeling and embedded conditional entropy coding of wavelet coefficients for image compression. In Proceedings of the 31st Asilomar Conference Record of the Thirty-First Asilomar Conference on Signals, Systems and Computers (Cat. No. 97CB36136), Pacific Grove, CA, USA, 2–5 November 1997; pp. 1378–1382.
4. ISO/IEC 15444-1:2019. Information technology—JPEG 2000 Image Coding System—Part 1: Core Coding System. 2019. Available online: https://www.iso.org/standard/78321.html (accessed on 11 November 2020).
5. Taubman, D. High Performance Scalable Image Compression with EBCOT. *IEEE Trans. Image Process.* **2000**, *9*, 1158–1170. [CrossRef] [PubMed]
6. Deever, A.; Hemami, S.S. What's your sign?: Efficient sign coding for embedded wavelet image coding. In Proceedings of the DCC 2000 Data Compression Conference, Snowbird, UT, USA, 28–30 March 2000; pp. 273–282.
7. Bilgin, A.; Sementilli, P.; Marcellin, M.W. Progressive image coding using trellis coded quantization. *IEEE Trans. Image Process.* **1999**, *8*, 1638–1643. [CrossRef] [PubMed]
8. Tian, C.; Hemami, S.S. An embedded image coding system baed on tarp filter with classification. In Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing, Montreal, QC, Canada, 17–21 May 2004.
9. Jerome, M.; Ellouze, N. Optimal image compression based on sign and magnitude coding of wavelet coefficients. *Int. J. Signal Process.* **2006**, *3*, 243–251.
10. Pearlman, W.A. Trends of tree-based, set partitioning compression techniques in still and moving image systems. In Proceedings of the 22nd Picture Coding Symposium: PCS-2001, Seoul, Korea, 25–27 April 2001; pp. 1–8.
11. Oliver, J.; Malumbres, M.P. Low-Complexity Multiresolution Image Compression Using Wavelet Lower Trees. *IEEE Trans. Circuits Syst. Video Technol.* **2006**, *16*, 1437–1444. [CrossRef]
12. Cho, Y.; Pearlman, W.A. Hierarchical Dynamic Range Coding of Wavelet Subbands for Fast and Efficient Image Compression. *IEEE Trans. Image Process.* **2007**, *16*, 2005–2015. [CrossRef] [PubMed]

13. Guo, J.; Mitra, S.; Nutter, B.; Karp, T. A Fast and Low Complexity Image Codec based on Backward Coding of Wavelet Trees. In Proceedings of the Data Compression Conference (DCC'06), Snowbird, UT, USA, 28–30 March 2006.
14. Zandi, A.; Allen, J.D.; Schwartz, E.L.; Boliek, M. CREW: Compression with reversible embedded wavelets. In Proceedings of the DCC '95 Data Compression Conference, Snowbird, UT, USA, 28–30 March 1995; pp. 212–221.
15. Deever, A.; Hemami, S.S. Efficient Sign Coding and Estimation of Zero-Quantized Coefficients in Embedded Wavelet Image Codecs. *IEEE Trans. Image Process.* **2003**, *12*, 420–431. [CrossRef] [PubMed]
16. Mallat, S.; Zhong, S. Characterization of Signals from Multiscale Edges. *IEEE Trans. Pattern Anal. Mach. Intell.* **1992**, *14*, 710–732. [CrossRef]
17. Martí, A.; López-Granado, O.; Rodríguez-Ballester, F.; Malumbres, M. Applying a Genetic Algorithm Solution to Improve Compression of Wavelet Coefficient Sign. In Proceedings of the 13th International Work-Conference on Artificial Neural Networks, IWANN, Mallorca, Spain, 10–12 June 2015; pp. 276–286. [CrossRef]
18. López-Granado, O.; Galiano, V.; Martí, A.; Migallón, H.; Martıínez-Rach, M.; Piñol, P.; Malumbres, M.P. Improving image compression through the use of evolutionary computing algorithms. *Data Manag. Secur. Appl. Med. Sci. Eng.* **2013**, 37–46. [CrossRef]
19. Navarro, M.J.; Moreno, P.; Rodríguez-Ballester, F.; Martí-Campoy, A.; Cruz Chávez, M.; Malumbres, M.; López Granado, O. Efficient Wavelet Sign Prediction: Simulated Annealing vs Genetic Algorithms. In Proceedings of the International Conference on Knowledge-Based and Intelligent Information and Engineering Systems (KES), San Sebastian, Spain, 10–12 September 2012; Volume 243, pp. 1676–1684. [CrossRef]
20. Bjontegaard, G. Document VCEG-M33: Calculation of Average PSNR Differences between RD-Curves. In Proceedings of the ITU-T VCEG Meeting, Austin, TX, USA, 2–4 April 2001.
21. Riordan, J. *Introduction to Combinatorial Analysis*; Princeton University Press: Princeton, NJ, USA, 1958.
22. Lourenço, H.; Martin, O.; Stützle, T. *Handbook of Metaheuristics*; Chapter Iterated Local Search; Springer: New York, NY, USA, 2019; pp. 129–168.
23. Papadimitriou, C.; Steiglitz, K. *Combinatorial Optimization: Algorithms and Complexity*; Dover Publications: Mineola, NY, USA, 1998.
24. El-Ghazali, T. *Metaheuristics form Design to Implementation*; Wiley: Hoboken, NJ, USA, 2009.
25. Cruz-Chávez, M.; Martínez-Oropeza, A.; Serna, B. Neighborhood Hybrid Structure for Discrete Optimization Problems. In Proceedings of the 2010 IEEE Electronics, Robotics and Automotive Mechanics Conference, Morelos, Mexico, 28 September–1 October 2010; pp. 108–113.
26. Kodak Lossless True Color Image Suite. Available online: http://r0k.us/graphics/kodak/ (accessed on 11 November 2020).
27. Rawzor—Lossless Compression Software for Camera Raw Images. Available online: http://imagecompression.info/test_images/ (accessed on 11 November 2020).

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.