Adaptive Wavelet Transform for Lossless Compression using Genetic Algorithm

Yasuo Takehisa
Information Technology Research Institute of Nagano Prefecture
1-7-7, Nomizo-nishi, Matsumoto, Nagano 399-0006, Japan
Phone: +81-263-25-0790
takehisa@nagano-it.go.jp

Hidenori Sakanashi
Electrotechnical Laboratory
1-1-4, Umezono, Tsukuba, Ibaraki
305-8568, Japan
Phone: +81-298-54-5883
sakanasi@etl.go.jp

Tetsuya Higuchi
Electrotechnical Laboratory
1-1-4, Umezono, Tsukuba, Ibaraki
305-8568, Japan
Phone: +81-298-54-5868
higuchi@etl.go.jp

Abstract

This paper proposes the adaptive wavelet transform for lossless (reversible) data compression using genetic algorithm (GA). In the proposed method, the lifting scheme (LS), which is the latest implementation method of wavelet transformation, is adopted for lossless compression, and GA optimizes the prediction mechanism used in LS according to the characteristics of the target image data, in order to achieve high compression ratio. The computer simulations demonstrate that the proposed method exhibits 9.8% better accuracy in prediction of pixel values in the target images, and it leads 1.9% better entropy of the transformed images than the conventional LS on the average.

1 INTRODUCTION

The *wavelet coding method* has been recognized as an efficient coding technique for lossy compression. The wavelet transform decomposes a typical image data to a few coefficients with large magnitude and many coefficients with small magnitude. Since most of the energy of the image concentrates on these coefficients with large magnitude, lossy compression systems just using coefficients with large magnitude can realize both high compression ratio and the reconstructed image with good quality at the same time. Wavelet transforms are now under the consideration as a part of JPEG2000, the next international standard for color image compression, because of its ability for effective concentration of the energy of an image (Charrier, 1999). However, applying wavelet transforms to lossless compressions create two problems. One is a fact that resulting transformed outputs are no longer integers. This problem can be solved by introducing *lifting scheme* (LS), one type of a spatial domain construction of wavelets which can map integers to integers easily (Calderbank, 1998). The another is achieving high compression ratio in lossless systems is difficult. Because lossless systems basically use all coefficients with not only large magnitude but also small magnitude which are usually neglected in lossy systems.

In this paper we give a solution to this problem and also propose a new preliminary method to apply wavelet transforms to lossless compression effectively. We adopt lifting scheme (LS) which is a new framework to construct wavelets in a spatial domain (Sweldens, 1997). We choose LS, the best method we think, which conducts wavelet transforms without any difficulties in mapping integers to integers required in lossless image compression systems (Calderbank, 1998). However, in terms of compression efficiency, the LS is not good in an effective compression, for the conventional LS has only standard prediction functions to cope with wide range real-world signals. Therefore, we have considered a method of changing prediction functions for each image to improve the compression ratio, gaining a more accurate prediction of pixels in the image.

The goal of this paper is to propose such an adaptive LS system using the genetic algorithm (GA) (Holland, 1975). Specifically, GA is used to improve the accuracy of the prediction functions in LS. The proposed method exhibits 9.8% better accuracy in prediction, and can reduce the entropy 1.9% better than the conventional prediction function on the average.

This paper is organized as follows: In section 2, the overview of LS is explained. Section 3 proposes the extended prediction function with GA in LS. In section 4, the performance of the proposed method is discussed through the results of computational simulations. Section 5 mentions two problems remained to be solved in the future and section 6 concludes this paper.

2 LIFTING SCHEME

This section describes the overview of the lifting scheme, and explains its advantages for lossless data compression.

2.1 OVERVIEW OF LIFTING SCHEME

The *wavelet transform*, which decomposes a signal into constituent parts in the time-frequency domain, has been successful in providing high compression ratios while maintaining good image quality (Antonini, 1992). It is well known that the wavelets can execute compression with a higher signal-to-noise ratio than DCT (discrete cosine transform) in the case of low bit-rate. Then, JPEG2000, the next international standard, is being developed based on wavelet transforms. However, the conventional wavelet transform needs additional mechanisms to execute lossless compression, of which recovered data is identical to the original one.

In contrast, the *lifting scheme* (LS), which is the latest implementation method of wavelet transform, doesn't have such problems and is very suitable for the lossless compression (Calderbank, 1998). It can decorrelate the target data in the space domain without Fourier transform (Sweldens, 1996). The considerable advantages of LS are,

(a) simple and fast procedure,

(b) ease of treating integer number, and

(c) ease of obtaining inverse transform.

The advantage (a) means that LS is exceedingly suitable for hardware implementation, because it uses only addition and multiplication, and requires small memory for calculations.

From the viewpoint of (b), the conventional wavelet transform has a problem in mapping integers to integers. It needs an additonal mechanism to cancel the rounding error for lossless compression. LS is, on the other hand, feasible to lossless compression, because it does not require such mechanisms to treat integer data.

The last advantage (c) makes LS useful in practical implementation. In the next subsection, LS is described in detail.

2.2 PROCEDURE OF LIFTING

The procedure of LS consists of three steps: *Split*, *Predict*, and *Update* (Figure 1).

*Split***:** Split the signal into two disjoint subsets of samples. We divide the original signal $x[n]$ into even and odd components: $x_e[n]$ and $x_o[n]$, where $x_e[n] = x[2n]$ and $x_o[n] = x[2n+1]$.

*Predict***:** Generate the detail signals $d[n]$ as the prediction error using a prediction operator P:

$$d[n] = x_o[n] - P(x_e[n])\,. \tag{1}$$

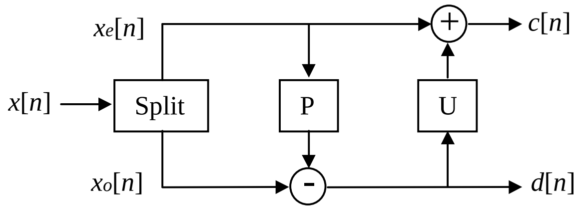

Figure 1: Lifting scheme

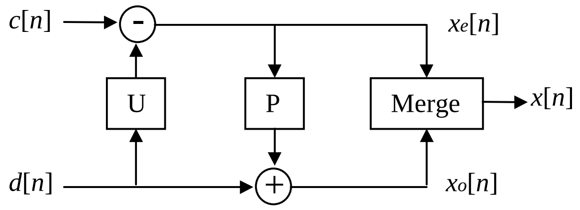

Figure 2: Inverse lifting scheme

*Update***:** Generate the coarser signals $c[n]$ by applying an update operator U to $d[n]$ and adding the result to $x_e[n]$:

$$c[n] = x_e[n] + U(d[n])\,. \tag{2}$$

They represent a coarse approximation to the original signal $x[n]$.

These three operations can be applied to $c[n]$ repeatedly. Moreover, the important point to note is that, in contrast to the conventional wavelet transform, we can easily obtain the *inverse lifting scheme* of any combination of prediction P and update U (Figure 2). From the equation (1) and (2), the $x_e[n]$ and $x_o[n]$ can be calculated from $c[n]$ and $d[n]$ as shown in the next equations:

$$x_o[n] = d[n] + P(x_e[n])\,, \tag{3}$$

$$x_e[n] = c[n] - U(d[n])\,. \tag{4}$$

This is the reason of the advantage (c) mentioned in the previous subsection. In this paper, using these advantages, the prediction function P is optimized by genetic algorithm according to the characteristics of the image

2.3 LIFTING SCHEME ON TWO DIMENSIONAL IMAGE DATA

When LS transforms the two dimensional data, the procedure explained in the previous subsection is applied two times (Figure 3).

At the first step, each row of the image is transformed and divided into two subimages, the sets of $\{c[n]\}$ and $\{d[n]\}$. Secondly, each subimage is scanned vertically, and their columns are transformed similarly.

As a result, the original image data is divided into four subimages, *cc*, *cd*, *dc* and *dd* in Figure 3. The subimage *cc* can be repeatedly transformed by LS, and this repetition is called *multi-resolution analysis*.

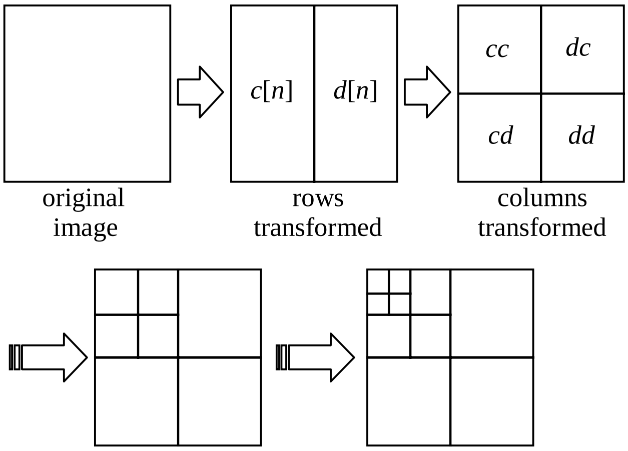

Figure 3: Multi-resolution sequential transformation

3 PROPOSED METHOD

In order to improve the performance of LS in lossless compression, we focus an attention on the accuracy of the prediction function P defined in equation (1). The proposed method improves the prediction accuracy by the following features:

- Development of the new prediction function (two dimensional configuration of the reference pixels),

- Employment of different prediction functions according to the local pattern of pixels (grouping pixel patterns into some categories), and

- Optimization of the coefficients in the prediction functions by means of genetic algorithm (GA).

In the next three subsections, these features are explained in turn.

3.1 TWO DIMENSIONAL CONFIGURATION OF REFERENCE PIXELS

Since the conventional LS is based on one dimensional signal processing, it does not make good use of two dimensional signals such as image data. That is, at the prediction of the pixels in $x_o[n]$, only the pixels in $x_e[n]$ on the same line are allowed to be used in the prediction function.

The configuration of reference pixels of typical LS is illustrated in Figure 4, in which $x_o^m[n]$ and $x_e^m[n]$ indicate the pixels at the odd and even positions in the m'th line of the image data, and the hatched pixels are allowed to be referred for predicting $x_o^m[n]$.

In the case of the most simple prediction, the pixel $x_e^m[n]$ and $x_e^m[n+1]$ are used for predicting $x_o^m[n]$. Generally, it would seem that the large number of reference pixels leads to the more accurate prediction, but $x_e^m[n-1]$ and $x_e^m[n+2]$, the second closest pixels from $x_o^m[n]$, don't contribute to the accurate prediction so much, because they are too distant from $x_o^m[n]$ to correlate with it strongly.

To overcome the difficulty, the proposed method arranges the reference pixels in two dimensional configuration as shown in the Figure 5. The additional reference pixels, $x_e^{m-1}[n]$, $x_e^{m-1}[n+1]$, $x_e^{m+1}[n]$, $x_e^{m+1}[n+1]$, are expected to correlate with $x_o^m[n]$ better than $x_e^m[n-1]$ and $x_e^m[n+2]$, because they are closer to $x_o^m[n]$.

Using the extended configuration of the reference pixels, the prediction function of the equation (1) is redefined as:

$$P(x_e[n]) = \sum_{k=-1}^{1}\sum_{l=0}^{1}\{p_{k,l}\, x_e^{m+k}[n+l]\}, \qquad (5)$$

where $p_{k,l}$ is the coefficient.

The next two subsections explain the relation between the coefficients and the method to optimize them according to the characteristics of the image, respectively.

$x_e^m[n-1]$		$x_e^m[n]$	$x_o^m[n]$	$x_e^m[n+1]$		$x_e^m[n+2]$

Figure 4: Configuration of reference pixels of conventional lifting scheme

$x_e^{m-1}[n]$		$x_e^{m-1}[n+1]$
$x_e^m[n]$	$x_o^m[n]$	$x_e^m[n+1]$
$x_e^{m+1}[n]$		$x_e^{m+1}[n+1]$

Figure 5: Two dimensional configuration of reference pixels

3.2 GROUPING OF PIXEL PATTERNS

The proposed method employs different prediction functions according to the pixel patterns neighboring the target pixels. In this subsection, we explain how the pixel patterns are categorized, and what prediction function is applied to each category.

We assume that the relations between coefficients in figure 5 are defined as:

$$\begin{aligned} &\sum_k \sum_l p_{k,l} = 1, \\ &p_{0,0} = p_{0,1}, \\ &p_{-1,0} = p_{1,0} = p_{-1,1} = p_{1,1}. \end{aligned} \tag{6}$$

In these equations, $p_{0,0}$ is the only variable, and the other coefficients are calculated using it as follows:

$$\begin{aligned} &p_{0,1} = p_{0,0}, \\ &p_{-1,0} = p_{1,0} = p_{-1,1} = p_{1,1} = \frac{1-2p_{0,0}}{4}. \end{aligned} \tag{7}$$

These are the basic relations between the coefficients, but we cannot use it at the boundaries of the image, because not all reference pixels are available. Then, some different prediction functions are prepared, and one of them is selected according to the positional relation between $x_o^m[n]$ and the boundaries. To improve the accuracy moreover, we change the coefficient $p_{0,0}$ according to the patterns of pixel values.

3.2.1 Classification based on Positional Relation between Pixels and Boundaries

We classify all probable pixel patterns into six cases as shown in Figure 6. In the figures, thick lines indicate the boundaries of the image, and x indicates the position of $x_o^m[n]$, the target pixel to be predicted.

In the case (a), based on the equation (6), the relation between the coefficients in the prediction function is defined as follows:

$$\begin{aligned} &\sum_k \sum_l p_{k,l} = 1, \\ &p_{0,0} = p_{0,1}, \\ &p_{1,0} = p_{1,1}, \\ &p_{-1,0} = p_{-1,1} = 0. \end{aligned} \tag{8}$$

Then, if the coefficient $p_{0,0}$ is determined, all other coefficients are calculated as,

$$\begin{aligned} &p_{0,1} = p_{0,0}, \\ &p_{-1,0} = p_{-1,1} = 0, \\ &p_{1,0} = p_{1,1} = \frac{1-2p_{0,0}}{2}. \end{aligned} \tag{9}$$

Similarly, the coefficients of the case (b) are

$$\begin{aligned} &p_{-1,1} = p_{0,1} = p_{1,1} = 0, \\ &p_{-1,0} = p_{1,0} = \frac{1-p_{0,0}}{2}, \end{aligned} \tag{10}$$

and the coefficients of the case (c) are

$$\begin{aligned} &p_{0,1} = p_{0,0}, \\ &p_{1,0} = p_{1,1} = 0, \\ &p_{-1,0} = p_{-1,1} = \frac{1-2p_{0,0}}{2}. \end{aligned} \tag{11}$$

The prediction accuracy in the cases (d) and (e) hardly influences the compression ratio, because these cases occur at only the corners of the image. Then, we fix their coefficients without optimization. The coefficients of the case (d) are

$$\begin{aligned} &p_{1,0} = 0.25, \\ &p_{0,0} = 0.75, \end{aligned} \tag{12}$$

and the ones of (e) are

$$\begin{aligned} &p_{0,0} = 0.75, \\ &p_{-1,0} = 0.25. \end{aligned} \tag{13}$$

The prediction function of case (f) is already defined in the equation (7), but we further classify it according to the pattern of the pixel values, as mentioned below.

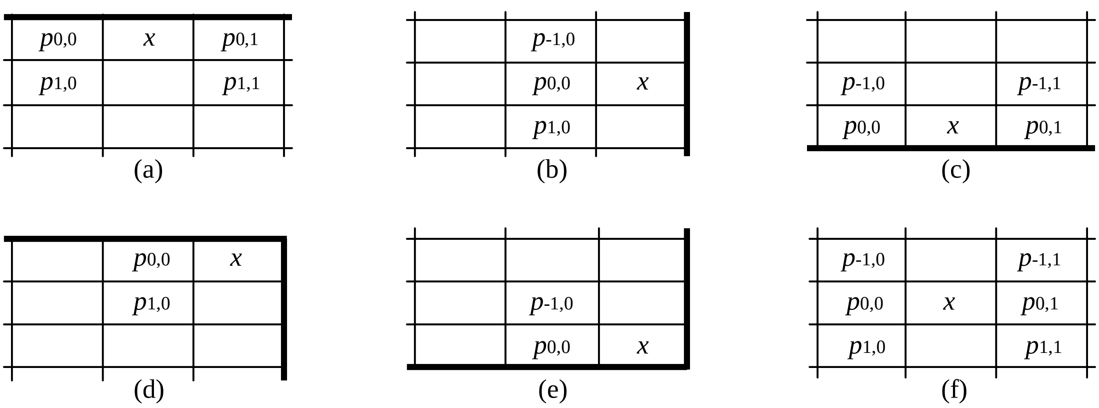

Figure 6: Patterns of relation between target pixel and boundaries of image

3.2.2 Classification According to Pattern of Pixel Values

Only the case (f) is classified into the four categories according to the correlation among the pixels neighboring the target pixel to be predicted, such as (i) Flat correlation, (ii) Horizontally flat correlation, (iii) Vertically flat correlation, and (iv) No correlation.

By using the different set of coefficients in each case, the pixel $x_o^m[n]$ can be predicted more accurately.

(i) Flat Correlation

In Figure 5, if the pixels neighboring the target pixel $x_o^m[n]$ satisfy the next equations, they seem to have very similar values.

$$\begin{aligned} &\left| x_e^{m-1}[n] - x_e^m[n] \right| < T \\ &\left| x_e^m[n] - x_e^{m+1}[n] \right| < T \\ &\left| x_e^{m-1}[n+1] - x_e^m[n+1] \right| < T \\ &\left| x_e^m[n+1] - x_e^{m+1}[n+1] \right| < T \\ &\left| x_e^{m-1}[n] - x_e^{m-1}[n+1] \right| < T \\ &\left| x_e^m[n] - x_e^m[n+1] \right| < T \\ &\left| x_e^{m+1}[n] - x_e^{m+1}[n+1] \right| < T \end{aligned} \quad (14)$$

(ii) Horizontally Flat Correlation

The next equations detect the horizontal flatness in the neighborhood of $x_o^m[n]$.

$$\begin{aligned} &\left| x_e^{m-1}[n] - x_e^{m-1}[n+1] \right| < T \\ &\left| x_e^m[n] - x_e^m[n+1] \right| < T \\ &\left| x_e^{m+1}[n] - x_e^{m+1}[n+1] \right| < T \end{aligned} \quad (15)$$

In this case, the coefficients are fixed as defined in the next equations because it is expected that only the horizontally adjacent pixels correlate to $x_o^m[n]$.

$$\begin{aligned} &p_{0,0} = p_{0,1} = 0.5 \\ &p_{-1,0} = p_{1,0} = p_{-1,1} = p_{1,1} = 0 \end{aligned} \quad (16)$$

(iii) Vertically Flat Correlation

When the reference pixels of $x_o^m[n]$ have the relation shown in the next equation, the vertical correlation is strong in the local area.

$$\begin{aligned} &\left| x_e^{m-1}[n] - x_e^m[n] \right| < T \\ &\left| x_e^m[n] - x_e^{m+1}[n] \right| < T \\ &\left| x_e^{m-1}[n+1] - x_e^m[n+1] \right| < T \\ &\left| x_e^m[n+1] - x_e^{m+1}[n+1] \right| < T \end{aligned} \quad (17)$$

Table 1: Summary of classification

	(i)	(ii)	(iii)	(iv)
(a)	Eqn.(9)			
(b)	Eqn.(10)			
(c)	Eqn.(11)			
(d)	$p_{0,0} = 0.75$, $p_{1,0} = 0.25$			
(e)	$p_{-1,0} = 0.25$, $p_{0,0} = 0.75$			
(f)	Eqn. (7)	$p_{0,0} = p_{0,1} = 0.5$ $p_{-1,0} = p_{1,0} = p_{-1,1} = p_{1,1} = 0$	Eqn. (7)	Eqn. (7)

(iv) No Correlation

If the reference pixels don't satisfy the above three conditions, the pixels neighboring $x_o^m[n]$ have no special correlation between them.

Table 1 summarizes the results of the above two types of classification. In the table, the phrase “Eqn. (y)” means that, in the corresponding case, the relation between the coefficient $p_{0,0}$ and the others are defined in equation y, and $p_{0,0}$ has to be determined as shown in the next subsection. That is, the proposed method requires seven parameters (six $p_{0,0}$'s and one threshold T) to execute one dimensional wavelet transform.

As mentioned in section 2.3, when the conventional LS executes multi-resolution analysis of the image, it uses the same prediction function three times for horizontal transformation of the original image, vertical transformation of the subimage $\{c[n]\}$, and vertical transformation of the subimage $\{d[n]\}$. By contrast, to achieve a better compression ratio, the proposed method uses three different prediction functions for three executions of one dimensional transformation. It means that the proposed method requires 7 × 3 = 21 parameters to process one image, since it needs seven parameters for one dimensional wavelet transform.

3.3 DECISION OF COEFFICIENTS BY GENETIC ALGORITHM

To optimize the 21 parameters, 18 $p_{0,0}$’s and 3 T’s, introduced in the previous subsection, we use the genetic algorithm, which is the powerful search procedure inspired from the adaptation of natural organisms (Holland, 1975). It has a population of 30 chromosomes, each of them representing a set of parameters. In this paper, the length of the chromosomes is 168 (Each parameter is represented in 8 bits).

At the initial state, the population is generated at random, and the chromosomes are evaluated to calculate the fitness value. The evaluation function F used in this paper is defined as follows:

$$F = -\sum_{m=1}^{H}\sum_{n=1}^{W}\{x_o^m[n] - \sum_{k=-1}^{1}\sum_{l=0}^{1}\{p_{k,l}\, x_e^{m+k}[n+l]\}\}, \quad (18)$$

where H and W represent height and width of the image data respectively, and the coefficients of p are determined by the chromosome itself. This function sums up the prediction error over all the image and returns the total value with a negative sign.

The genetic operations and GA parameters used in the proposed method are shown in Table 2. After evaluation, the chromosomes are processed by three genetic operators: the tournament selection, the uniform crossover, and the point mutation. The set of evaluation and genetic operations are repeated 100 times.

Table 2: Genetic operations and parameter setting of GA

Genetic operation	
GA model	Elitist simple GA
Selection	Tournament
Crossover	One point
Parameter setting of GA	
Total generation	100
Population size	100
Gene length	8
Tournament size	10
Crossover ratio	0.8
Mutation ratio	0.03

4 COMPUTATIONAL SIMULATION AND DISCUSSION

To evaluate the performance, the proposed system is applied to six test images contained in SIDBA (Standard Image Data BAse), as shown in Figure 7. The two types of prediction function in the conventional LS are also applied to the same test images. They use the following prediction functions (Fernandez, 1996):

$$P(x_e[n]) = \frac{1}{2} x_e^m[n] + \frac{1}{2} x_e^m[n+1] , \qquad (19)$$

$$P(x_e[n]) = \frac{-x_e^m[n-1] + 9x_e^m[n] + 9x_e^m[n+1] - x_e^m[n+2]}{16} . \qquad (20)$$

These two equations are called “linear prediction” and “cubic prediction”, respectively. Also, we apply hill-climb instead of GA to evaluate how much proposed system is done by GA than the other methods.

Table 3 shows the prediction errors of the conventional LS, hill-climb and the proposed method. This table tells us three facts: First, the cubic prediction exhibits worse performance than the linear prediction, even though it uses more reference pixels. It means that the pixels distant from the target pixel don’t contribute to the prediction accuracy. This fact supports evidence that two dimensional configuration of the reference pixel (proposed in section 3.1) is effective to improve the prediction accuracy.

Second, the proposed method exhibits better performance than the conventional LS in all test images. Then, it is expected that the proposed method may work well on a wide variety of images. In addition, the small prediction error caused by the accurate prediction is efficient in lossy coding as well as lossless coding.

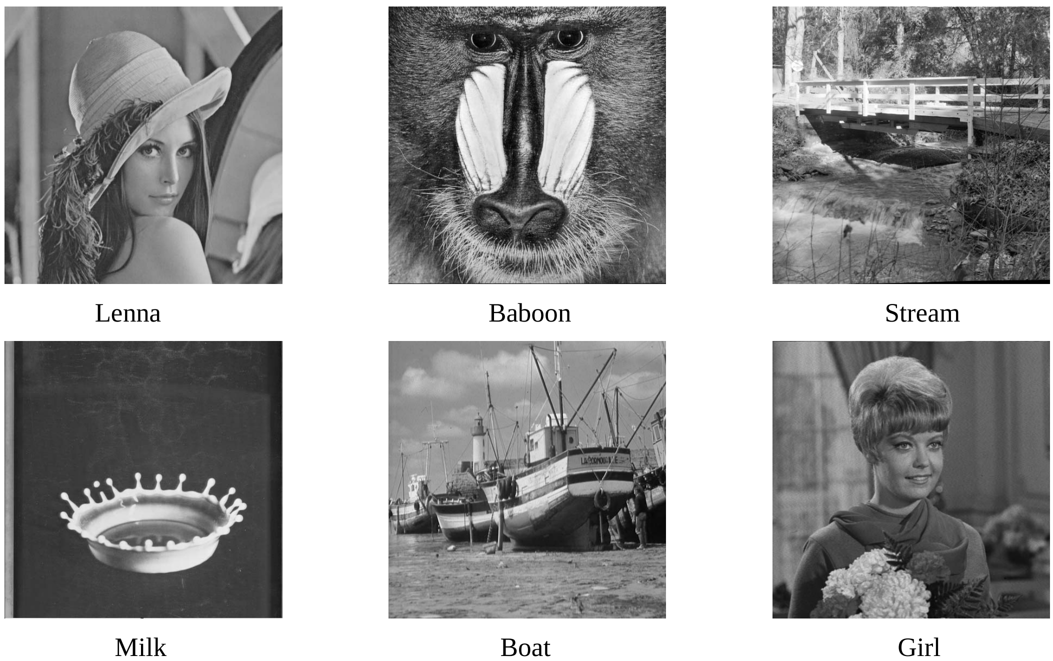

Figure 7: Test images. All images are 512 × 512.

Table 3: Prediction error of conventional methods, hill-climb and proposed method

Image	Conventional Method		Hill-Climb	Proposed Method			
	Linear	Cubic			Improvement(%)		
					from Linear	from Cubic	from H-C.
Lenna	579490	586356	570252	553036	104.8	106.0	103.1
Baboon	2707202	2819412	2704855	2654867	102.0	106.2	101.9
Stream	1745753	1819541	1741177	1713726	101.9	106.2	101.6
Milk	548860	583705	532643	512408	107.1	113.9	103.9
Boat	1120762	1149974	1070015	1033743	108.4	111.2	103.5
Girl	823559	863326	761741	748876	110.0	115.3	101.7
Average					105.7	109.8	102.6

Table 4: Entropy of images transformed by conventional methods, hill-climb, and proposed method

Image	Conventional Method		Hill-Climb	Proposed Method			
	Linear	Cubic			Improvement(%)		
					from Linear	from Cubic	from H-C.
Lenna	3.881	3.895	3.870	3.846	100.9	101.3	100.6
Baboon	5.031	5.057	5.042	5.017	100.3	100.8	100.5
Stream	4.305	4.439	4.377	4.367	98.6	101.7	100.2
Milk	3.580	3.634	3.550	3.522	101.7	103.2	100.8
Boat	4.332	4.351	4.297	4.269	101.5	101.9	100.6
Girl	4.138	4.174	4.082	4.069	101.7	102.6	100.3
Average					100.8	101.9	100.5

Third, when we compare GA with hill-climb, GA's performances exceed hill-climb's ones over all images. This fact tells us that GA works better than hill-climb for optimizing parameters in this problem.

Table 4 shows the entropy of the test images transformed by the conventional LS, hill-climb, and the proposed method. In almost all images, the proposed method succeeds to reduce the entropy, but the ratio of improvement is very small compared with improvement in the prediction error. On the contrary, in "Stream", the entropy of the proposed method becomes worse than the linear prediction. This deterioration originates in the fact which is mentioned below. Since the evaluation function defined in equation (18) does not concern the entropy, there is no guarantee of improvement in the entropy. Then, the evaluation function of GA needs to be modified to obtain a better entropy.

5 FUTURE DEVELOPMENTS

We still leave two problems to be considered in the proposed method. First, although we evaluate only prediction functions, the lifting scheme should be evaluated as an entire system including update operations. Second, since a nonlinear prediction function is used, the nonlinearity is propagated to the low frequency component, c[n] in Figure 1, which is the target of the following lifting scheme. The nonlinearity would be accumulated in the low frequency component while implementing the multi-resolution analysis. It could influence bad effects for the image compression.

In the future, we will study the above two problems carefully and complete the whole adaptive lifting scheme.

6 CONCLUSION

In this paper, we proposed the method to optimize wavelet function for lossless data compression using genetic algorithm (GA). In the proposed method, the lifting scheme (LS), which is the latest implementation method of wavelet transform, is combined with GA. Specifically, GA optimizes the prediction function of LS according to the target image to be compressed. The computer simulation demonstrated that the proposed method presented 5.7% or 9.8% better accuracy than linear or cubic prediction function in prediction error, and reduced the entropy of the transformed image 0.8% or 1.9% better than the above two conventional functions.

References

M. Charrier, D. Santa Cruz, M. Larsson: JPEG2000, theNext Millenium Compression Standard for Still Images, Proc. of the IEEE ICMCS99, 1(2):pp.131-132 (1999).

W. Sweldens and P. Schróder: Building your own wavelets at home, In “Wavelets in Computer Graphics”, ACM SIGGRAPH Course Notes (1996).

W. Sweldens: The lifting scheme: A construction of second generation wavelets, Siam J. Math. Anal, Vol. 29, No. 2, pp 511-546 (1997).

J. H. Holland: Adaptation in Natural and Artificial Systems, The University of Michigan Press (1975).

M. Antonini, M. Barlaud, P. Mathieu, I. Daubechies: Image coding using wavelet transform, IEEE Trans. Image Processing, 1(2), pp.205-220 (1992).

R.C. Calderbank, I. Daubechies, W. Sweldens, and B. Yeo: Wavelet Transforms that Map Integers to Integers, Applied and Computational Harmonic Analysis (ACHA), Vol. 5, No. 3, pp. 332-369 (1998).

G. Fernandez, S. Periaswamy, and W. Sweldens: LIFTPACK: A Software Package for Wavelet Transforms using Lifting, Wavelet Applications in Signal and Image Processing IV, pp. 396-408 (1996).