

## Research Article

# An Evolved Wavelet Library Based on Genetic Algorithm

D. Vaithyanathan,<sup>1</sup> R. Seshasayanan,<sup>1</sup> K. Kunaraj,<sup>2</sup> and J. Keerthiga<sup>1</sup>

<sup>1</sup> Department of Electronics and Communication Engineering, Anna University, Chennai 600025, India

<sup>2</sup> Department of Electronics and Communication Engineering, Loyola-ICAM College of Engineering and Technology (LICET), Chennai 600034, India

Correspondence should be addressed to D. Vaithyanathan; [vaithi\\_d@rediffmail.com](mailto:vaithi_d@rediffmail.com)

Received 28 January 2014; Revised 31 July 2014; Accepted 28 August 2014; Published 27 October 2014

Academic Editor: Dimitrios A. Karras

Copyright © 2014 D. Vaithyanathan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As the size of the images being captured increases, there is a need for a robust algorithm for image compression which satiates the bandwidth limitation of the transmitted channels and preserves the image resolution without considerable loss in the image quality. Many conventional image compression algorithms use wavelet transform which can significantly reduce the number of bits needed to represent a pixel and the process of quantization and thresholding further increases the compression. In this paper the authors evolve two sets of wavelet filter coefficients using genetic algorithm (GA), one for the whole image portion except the edge areas and the other for the portions near the edges in the image (i.e., global and local filters). Images are initially separated into several groups based on their frequency content, edges, and textures and the wavelet filter coefficients are evolved separately for each group. As there is a possibility of the GA settling in local maximum, we introduce a new shuffling operator to prevent the GA from this effect. The GA used to evolve filter coefficients primarily focuses on maximizing the *peak signal to noise ratio* (PSNR). The evolved filter coefficients by the proposed method outperform the existing methods by a 0.31 dB improvement in the average PSNR and a 0.39 dB improvement in the maximum PSNR.

## 1. Introduction

Initially GA was developed to modify the coefficient sets of standard wavelet inverse transform which significantly improved the MSE for a given class of one-dimensional signals [1]. An investigation on evolutionary computation for image compression shows that it can be used to optimize wavelet coefficients and the transforms are independently trained and tested using three sets of images: digital photographs, fingerprints, and satellite images [2–4] and it was concluded that a better evolutionary progress towards an optimized reconstruction transform occurs when both the wavelet and scaling numbers are simultaneously evolved. Coevolutionary genetic algorithm based wavelet design for compressing fingerprint images was developed [5, 6] and the evolved wavelets outperform hand-design wavelet improving the quality of compressed images significantly. The suitability of the evolutionary strategy (ES) to implement it in Field Programmable Gate Array (FPGA) was investigated and the

original algorithm was modified by cutting down several computing requirements [7–9]. The discrete wavelet transform (DWT) coefficients evolved using GA showed better compression and reconstruction of images with less MSE compared to 9/7 wavelet [10] and the detrimental effects of quantization for ultrasound images are compensated using the evolved transforms and its superior performance increases in proportion to the selected quantization level [11]. Moore et al. evolved matched filter pairs for deep space images that outperformed standard wavelets [12]. Even at three-level multiresolution analysis (MRA) transforms the evolved filters gives better compression performance for both photographic [3, 13] and satellite images [14, 15]. The adaptive embedded system developed by Salvador et al. performs an adaptive image compression in FPGA devices and finds the optimized set of wavelet filters in less than 2 minutes when the input image changes [16, 17]. Recently an adaptive fingerprint image compression (FIC) technique was carried out by evolving optimized lifting coefficients [18]. Evolving

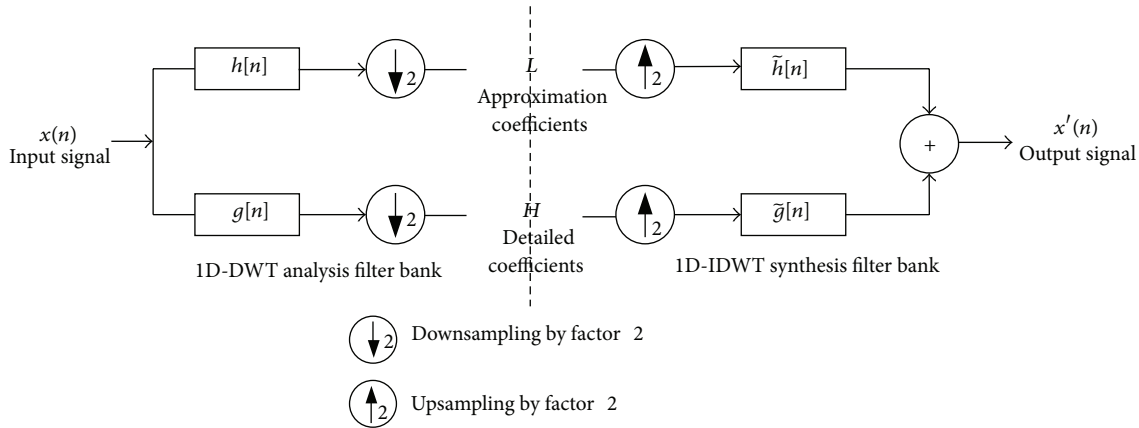


FIGURE 1: Single level wavelet transforms using convolution scheme.

DWT filter coefficients separately for near-edge pixels and far-edge pixels have proven significant improvement in error when the images are reconstructed. Isolation of edge pixels can be done by the conventional edge detection algorithms like Sobel detector and a corresponding binary mask will separate the image into near-edge and far-edge objects [9].

**1.1. Contribution.** Primarily the input images are classified based on their frequency content, calculated by performing the DWT, and the corresponding method is detailed in Section 3.2. The training images are grouped according to the calculated average frequency metric and for each group separate DWT filter coefficients are evolved. The fitness function is formulated using PSNR value only but in the future we would like to extend the fitness function as a combination of PSNR, *energy compaction* (EC), and *structural similarity* (SSIM) index [19]. Perhaps, the authors believe that the optimization of wavelet filter coefficients with multiobjective fitness function formulated using PSNR, EC, and SSIM would yield a set of filter coefficients with better compression performance [20]. In this paper, the authors work is limited for the evolution of a library of wavelet filter coefficients for various groups of images considering the PSNR as the fitness function.

The rest of the paper is organized as follows. Section 2 gives the sufficient background to understand the wavelets and genetic algorithm. Section 3 pursues with the image classification based on frequency content. Detailed experimental setup for evolving DWT filter coefficients and the analysis of quality metrics of the reconstructed images are discussed in Section 4. The paper is concluded in Section 5 with the possible enhancements.

## 2. Background

The main objective of this paper is to evolve wavelet filter coefficients suitable for image compression for various groups of images classified according to their spatial frequency

content. A detailed discussion about wavelets and genetic algorithm would be essential.

**2.1. Wavelets and Image Compression.** The wavelet is a multiresolution analysis tool widely used in signal and image processing. The analysis of the signal can be carried out at different frequencies and also with different time resolutions. It should be noted that there is a trade-off between frequency resolution and time resolution in wavelet. Hence the wavelet can be designed to provide good frequency resolution by giving off the time resolution and vice versa.

Discrete wavelet transforms (DWTs) are widely used for image compression as they have good compression capability. In particular, biorthogonal wavelets prove remarkable capabilities in still image compression. Perhaps the lifting scheme based DWT converts the high pass and low pass filtering operations into sequence of matrix multiplications and hence it proves to be efficient in terms of computation and memory.

**2.1.1. Discrete Wavelet Transform.** The wavelet decomposition of the signal into different frequency bands is simply obtained by successive high pass and low pass filtering of the time domain signal. The original input signal  $x[n]$  is first passed through a half band high pass filter  $g[n]$  and a low pass filter  $h[n]$ . After the filtering process, half of the samples can be eliminated according to the Nyquist rule. The signal now has a highest frequency of  $\pi/2$  radians instead of  $\pi$ . The signal  $x[n]$  can therefore be subsampled by 2, by discarding every other sample. This constitutes one level of wavelet decomposition as shown in Figure 1 and can mathematically be expressed as follows:

$$Y_{\text{High}}[k] = \sum_n x[n] \cdot g[2k - n] \quad (1)$$

$$Y_{\text{Low}}[k] = \sum_n x[n] \cdot h[2k - n].$$

The above procedure is followed in reverse order for the reconstruction. The signals are upsampled at every level and

TABLE 1: CDF 9/7 filter coefficients.

n	Analysis filter coefficients		Synthesis filter coefficients	
	Low	High	Lowr	Highr
1	0.02674875741080976	0.09127176311424948	0.09127176311424948	0.02674875741080976
2	-0.01686411844287495	-0.05754352622849957	-0.05754352622849957	0.01686411844287495
3	-0.07822326652898785	-0.5912717631142470	0.5912717631142470	-0.07822326652898785
4	0.2668641184428723	1.115087052456994	1.115087052456994	-0.2668641184428723
5	0.6029490182363579	-0.5912717631142470	0.5912717631142470	0.6029490182363579
6	0.2668641184428723	-0.05754352622849957	-0.05754352622849957	-0.2668641184428723
7	-0.07822326652898785	0.09127176311424948	-0.09127176311424948	-0.07822326652898785
8	-0.01686411844287495			0.01686411844287495
9	0.02674875741080976			0.02674875741080976

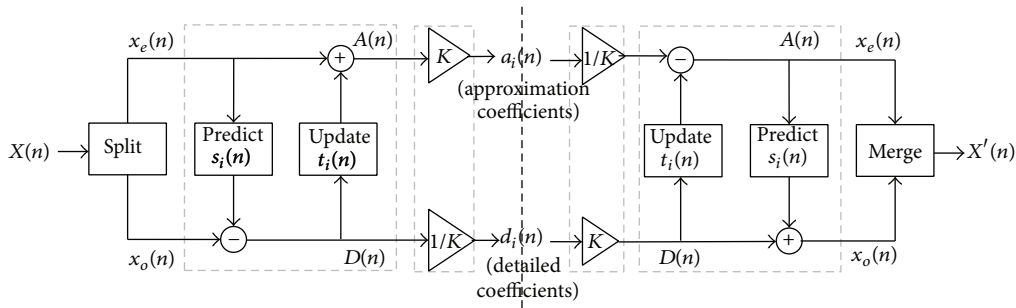


FIGURE 2: Discrete wavelet transforms using lifting scheme.

passed through the synthesis filters  $\tilde{g}[n]$  (high pass) and  $\tilde{h}[n]$  (low pass) and then added:

$$x'[n] = \sum_{k=-\infty}^{\infty} (Y_{\text{High}}[k] \cdot \tilde{g}[-n+2k]) + (Y_{\text{Low}}[k] \cdot \tilde{h}[-n+2k]). \quad (2)$$

Fast wavelet transform (FWT) and/or Mallat's herringbone algorithm [21] which is a computationally efficient implementation of the DWT is used here to compute the wavelet coefficients. Table 1 shows the CDF 9/7 filter coefficients for both forward and inverse DWT.

Wavelets are described by four sets of coefficients:

- (1) LOW is the set of wavelet numbers for the forward DWT,
- (2) HIGH is the set of scaling numbers for the DWT,
- (3) LOWR is the set of wavelet numbers for IDWT,
- (4) HIGHR is the set of scaling numbers for the IDWT.

**2.1.2. Lifting Based DWT and IDWT.** Lifting scheme is a computationally efficient way of implementing DWT [22, 23]. The transform can proceed first with the *Lazy Wavelet*, then alternating *dual lifting* and *primal lifting* steps, and finally a scaling. The inverse transform proceeds first with a scaling, then alternating lifting and dual lifting steps, and finally the inverse lazy transform. The inverse transform can immediately be derived from the forward transform by running the scheme backwards and flipping the signs.

The polyphase decomposition of discrete low pass (LOW(Z)) and high pass (HIGH(Z)) filters are

$$\begin{aligned} \text{LOW}(Z) &= \text{LOW}_e(Z^2) + Z^{-1}\text{LOW}_o(Z^2) \\ \text{HIGH}(Z) &= \text{HIGH}_e(Z^2) + Z^{-1}\text{HIGH}_o(Z^2). \end{aligned} \quad (3)$$

The synthesis filters can be expressed through polyphase matrix:

$$T(Z) = \begin{pmatrix} \text{LOW}_e(Z) & \text{HIGH}_e(Z) \\ \text{LOW}_o(Z) & \text{HIGH}_o(Z) \end{pmatrix}. \quad (4)$$

And  $\hat{T}(Z)$  can be analogously defined for the analysis filters. Euclidean algorithm can be used to decompose  $T(Z)$  and  $\hat{T}(Z)$  as

$$T(Z) = \prod_{i=0}^m \begin{bmatrix} 1 & P_i(Z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ U_i(Z) & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & \frac{1}{K} \end{bmatrix} \quad (5)$$

$$\begin{aligned} \hat{T}(Z^{-1})^{\text{transpose}} &= \prod_{i=0}^m \begin{bmatrix} 1 & 0 \\ -P_i(Z^{-1}) & 1 \end{bmatrix} \begin{bmatrix} 1 & -U_i(Z^{-1}) \\ 0 & 1 \end{bmatrix} \\ &\times \begin{bmatrix} \frac{1}{K} & 0 \\ 0 & K \end{bmatrix}. \end{aligned} \quad (6)$$

The discrete wavelet transform using lifting scheme consists of three steps as in Figure 2.

- (1) *Split*: the original signal,  $X(n)$ , is split into odd and even sequences (lazy wavelet transform)

$$X_e(n) = X(2n) \tag{7}$$

$$X_o(n) = X(2n + 1). \tag{8}$$

- (2) *Lifting*: it consists of one or more steps  $m$  of the form.

- (a) *Predict/dual lifting*: if  $X(n)$  possesses local correlation, then  $X_e(n)$  and  $X_o(n)$  also have local correlation; therefore, one subset (generally odd sequence) is used to predict the other subset (even sequence). Thus, the prediction step consists of applying a filter to the even samples and subtracting the result from the odd ones:

$$D(n) = X_o(n) - P[X_e(n)], \tag{9}$$

where  $P[X_e(n)]$  expresses that the value of  $D(n)$  is predicted by some combination of the value of  $X_e(n)$ .

- (b) *Update/primal lifting*: an update step does the opposite of applying a filter to the odd samples and adding the result from the even samples:

$$A(n) = X_e(n) + U[D(n)]. \tag{10}$$

Eventually, after  $m$  pairs of prediction and update steps, the even samples become the low pass coefficients while the odd samples become the high pass coefficients.

- (3) *Normalization/scaling*: after  $m$  lifting steps, scaling coefficients  $1/K$  and  $K$  are applied to the odd and even samples, respectively, in order to obtain the high pass subband ( $H$ ) and low pass subband ( $L$ ).

Lifting scheme for biorthogonal 9/7 is as follows.

*Lifting Steps*

$$\text{Predict } P1 : D_o(n) = X_o(n) + a[X_e(n) + X_e(n + 1)]$$

$$\text{Update } U1 : A_o(n) = X_e(n) + b[D_o(n - 1) + D_o(n)]$$

$$\text{Predict } P2 : D_1(n) = D_o(n) + c[A_o(n) + A_o(n + 1)]$$

$$\text{Update } U2 : A_1(n) = A_o(n) + d[D_1(n - 1) + D_1(n)]. \tag{11}$$

*Scaling*

$$X_L(n) = K \times A_1(n)$$

$$X_H(n) = \frac{1}{K} \times D_1(n). \tag{12}$$

$a = -1.586134342, b = -0.052980118, c = 0.882911075, d = 0.443506852,$  and  $k = 1.2301741049/1.149604398.$

Thus by adapting wavelets to better suit the image, the performance of image compression can be increased. This adaptation is done by an evolutionary algorithm (EA) such

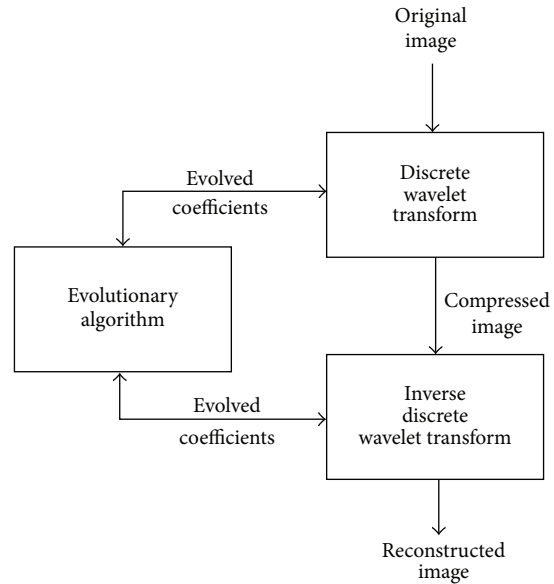


FIGURE 3: Evolutionary wavelet transform.

as GA to improve the image reconstruction in the presence of quantization error by replacing the wavelet filter coefficients with a set of evolved filter coefficients. Evolutionary algorithm will evolve the best filter coefficients for the given image as shown in Figure 3.

*2.2. Genetic Algorithm.* Genetic algorithms (GAs) (first proposed by Holland) have frequently been used to solve a number of difficult optimization problems. GAs work by first creating a population of randomly generated chromosomes. Over a number of generations, new chromosomes are created by mutating and recombining chromosomes from the previous generation. Among the total population, the best chromosomes (solutions) are then selected for survival to the next generation based on some fitness criteria. The flow diagram of the GA for evolving wavelet filter coefficients is shown in Figure 4.

- (a) Types of genetic algorithm.

- (i) Binary coded GA.
- (ii) Real coded genetic algorithm (RCGA).

*2.2.1. Real Coded Genetic Algorithm.* In RCGA the chromosomes are represented as real valued coefficients. The evolution of filters for image processing requires the simultaneous optimization of many real valued coefficients.

*Population Initialization.* The initial population includes one chromosome consisting of CDF9/7 filter coefficients. The remaining individuals are copies of the original wavelet filter coefficients multiplied by a small random factor. Additionally, 5% of the filter coefficients are negated. Each chromosome is composed of low pass filter coefficients, high pass filter coefficients, low pass filter reconstruction coefficients, and high pass filter reconstruction coefficients.

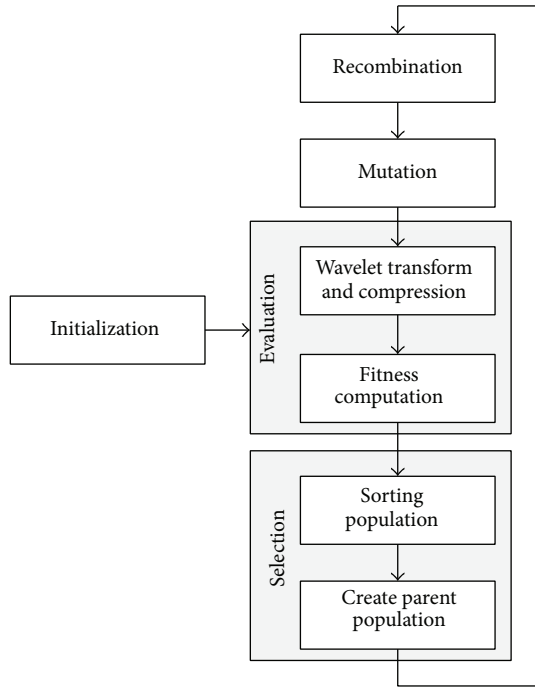


FIGURE 4: GA for evolving wavelet filter coefficients (Salvador et al., [7]).

*Evaluation.* The fitness of initial population is evaluated by first performing two-dimensional (2D) DWT on the test images and then the conventional decomposition and reconstruction (refer to Figure 5) is performed on the transformed coefficients and finally 2D IDWT is carried out to get the reconstructed image and the population is sorted according to the average fitness value.

Image quality (PSNR) and distortion (MSE) metrics are calculated between the original and the reconstructed image and the PSNR value is taken as the fitness measure. PSNR and MSE between the original ( $X$ ) and reconstructed ( $\hat{X}$ ) image of size  $M \times N$  can be calculated using (13) and (14), respectively. Here  $B$  represents bits per pixel (bpp):

$$\text{PSNR (dB)} = 20 \log_{10} \left( \frac{2^B - 1}{\sqrt{\text{MSE}}} \right), \quad (13)$$

$$\text{MSE} = \frac{\|X - \hat{X}\|^2}{MN}. \quad (14)$$

An  $\text{MSE} = 0$  in a reconstructed image indicates that  $\hat{X}$  is a perfect reconstruction of  $X$ . Increasing values of MSE correspond to increasing error.

*New Population Creation.* Once the population is evaluated for its performance, the new population is created from the parent population by the following.

- (i) *Sorting* the population according to the evaluated fitness measure.
- (ii) *Selecting* the parents for reproduction by random/stochastic uniform selection methods.
- (iii) *Reproducing* the population for the next generation.

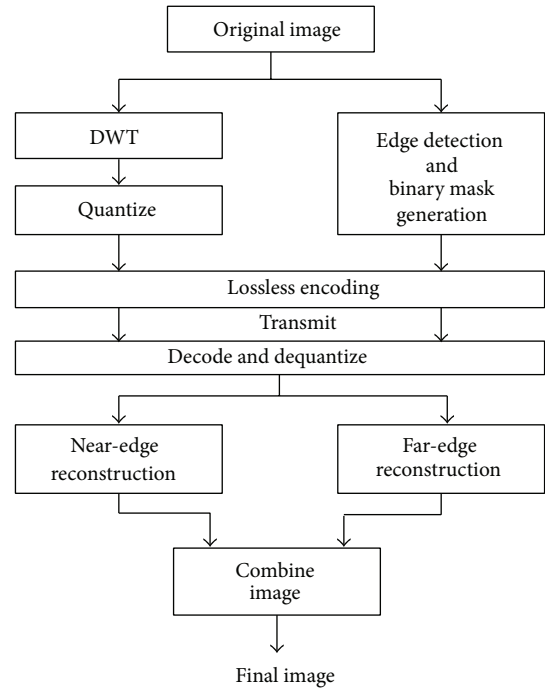


FIGURE 5: Image decomposition and reconstruction with evolved filters targeting edge-adjacent and non-edge-adjacent portions of image (Peterson and Lamont [9]).

*Reproduction (Recombination and Mutation).* The new population for next generation is created by crossover and mutation.

- (i) *Elite.* It represents the number of best individuals which is copied from the parent population to the new population;  $N_e$  is elite count number.
- (ii) *Heuristic Crossover.* The technique by which a child is created from two parents  $P_i^1$  and  $P_i^2$  biased in the direction of the parent with better fitness. Assuming  $P_i^1$  has better fitness than  $P_i^2$ , then a child gene  $C_i$  is created as

$$C_i = r(P_i^1 - P_i^2) + P_i^1, \quad (15)$$

where  $r$  is randomly chosen in the interval  $[0, 1]$ .

- (iii) *Gaussian Mutation.* Mutation is required to avoid the premature population convergence in RCGA. Given a parent vector  $P$ , a new child vector  $C$  is created by  $C = P + M$ ;  $M$  is based on Gaussian mutation, where the mutation shrinks in successive generations. Mutation Shrink rate controls the rate at which the average amount of mutation decreases. In early generations, the large variance permits quick exploration of the search space. Towards the end of the run, the variance is quite small, and the mutation operator makes very small refinements. If  $k$  is the current generation,



FIGURE 6: Relationship between detail subbands and image content.



FIGURE 7: Test images.

“gens” is the total number of generations in the GA run. Thus the variance is calculated as

$$\text{var}_k = \text{var}_{k-1} \left( 1 - 0.75 * \left( \frac{k}{\text{gens}} \right) \right). \quad (16)$$

*Proposed Shuffling Mechanism.* The probability of occurrences of the optimum solution increases by increasing the

number of generation runs. The shuffling mechanism is primarily introduced to avoid the search algorithm getting struck at a local minimum [24]. Perhaps the search algorithm sometimes settles to the local minimum point and we call this phenomenon “positional effect” which can be avoided using the proposed “shuffling mechanism.” This shuffling operator totally changes the position of the elite chromosomes while

getting replaced as new population for the next iteration. To a certain extent, this can make the search algorithm further visit some steepest points in the search space.

The proposed GA along with the genetic operators and shuffling mechanism is tested for convergence using few standard objective functions, namely, Rosenbrock function, De Jong's function, and Rastrigin's function and the results obtained show that the proposed GA is suitable for the optimization problem. The optimum solutions obtained by the proposed GA for the standard test functions are listed in Table 2.

**2.2.2. Genetic Algorithm Configuration for Evolving Global and Local Filters.** The overriding goal of this research work is to develop a robust methodology for the evolutionary optimization of image transform filters capable of outperforming CDF 9/7 under conditions subject to quantization noise.

*Evolving Global and Local Filters.* Traditional image transformation algorithms are concerned with minimizing the global error between a reconstructed image and its original counterpart. Those transforms which are evolved to provide reconstruction over entire images tend to exhibit higher error rates near image object edges (Salvador et al., [8]).

*Improved Reconstruction through Edge Detection and Targeted Evolution.* Thus to improve the reconstruction of edges within an image, the image is reconstructed using two evolved image filters, globally evolved filters (Filters evolved using the entire image for fitness calculation to reduce errors in areas not adjacent to object edges.) and locally evolved filters (evolved using the edge-enclosing masks for fitness calculation to reduce error near object edges) and the two reconstructed images are combined by using binary mask which is generated by edge detection (canny edge detector) followed by thresholding. Figure 5 describes the process involved.

Initially the algorithm starts with separating the image into near-edge pixels and far-edge pixels using edge detector algorithm. Once the edges are detected, a binary mask is created which is a binary image which carries black pixels in the far-edge area and white pixels in the near-edge image area based on the threshold value. Hence, there are two classes of images (near and far edge) through which the evolutionary algorithm separately evolves suitable filter coefficients for the given set of training images. DWT is taken for both images using the respective filter coefficients and then it is quantized and encoded using lossless encoding algorithm like Huffman coding and transmitted. In the receiving side, the image can be reconstructed using appropriate wavelet filters and the individual near-edge and far-edge images are combined together to form a complete image.

**2.3. Evolution of Wavelets.** Evolution of wavelets can be carried out in the following ways.

(i) Convolution scheme

- (1) 9 variables (L1, L2, L3, L4, L5, H1, H2, H3, and H4);
- (2) 2 variables (LR3 and LR4).

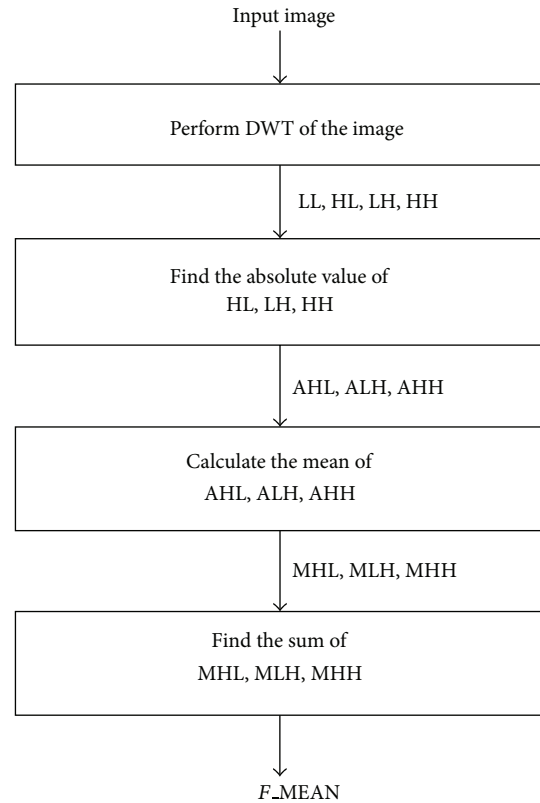


FIGURE 8: Calculation of *F.MEAN*.

(ii) Lifting scheme

- (1) 5 variables (*a, b, c, d,* and *k*);
- (2) 1 variable (*k*).

**2.3.1. Convolution Scheme**

*Population Initialization.* The initial population includes one chromosome consisting of CDF 9/7 filter coefficients. The remaining individuals are copies of the original wavelet coefficients multiplied by a small random factor. Additionally, 5% of the filter coefficients are negated. The initial configuration of the GA for each scheme is discussed in Table 3.

Each chromosome is composed of the following:

- (i) low pass filter coefficients (9);
- (ii) high pass filter coefficients (7);
- (iii) low pass filter reconstruction coefficients (7);
- (iv) high pass filter reconstruction coefficients (9).

Initial population consists of the following:

- (i) biorthogonal 9/7 (seed);
- (ii)  $n - 1$  copies of 9/7 multiplied by a small noise factor.

*Evolving 9 Filter Coefficients.* In this method there are no constraints regarding the evolution of filter coefficients. The filter coefficients are allowed to evolve randomly and it is

Group Images



FIGURE 9: Group of classified images.



TABLE 2: Optimum solutions obtained by the proposed GA.

Test function	Minimum value	Theoretical solution	Optimum solutions obtained by the proposed GA
Rosenbrock function $f(x, y) = 100(y - x^2)^2 + (1 - x^2)$ , $-2 < (x, y) < 2$	0.024433341	(1, 1) (0.896, 0.803)	$x = 0.9959105$ $y = 1.0074636$
De Jong's function in 2D $f(x, y) = x^2 + y^2$ , $-5 < (x, y) < 5$	0.847917595	(0, 0)	$x = 0.0863404$ $y = 0.0189192$
Rastrigin's function $f(x, y) = 10 \times 2 + [x^2 - 10 \cos(2\pi x)] + [y^2 - 10 \cos(2\pi y)]$ , $-5 < (x, y) < 5$	0.423719237	(0, 0)	$x = 0.00539928$ $y = 0.04605763$

TABLE 3: Initial configuration of four different GAs.

Parameter variables	Convolution filter		Lifting scheme	
	9	2	5	1
Wavelet DWT implementation	Matlab wavelet toolbox	Matlab wavelet toolbox	Matlab wavelet toolbox	Matlab wavelet toolbox
Fitness function	PSNR	PSNR	PSNR	PSNR
Generations	500	500	500	500
Permitted filter coefficient values	[21.0, 1.0]	[-0.2, 0.4] [-0.1, 0.3]	[-1.5, 1.5]	[1.0, 1.3]
Population initialization	CDF9/7	CDF9/7	CDF9/7	CDF9/7
Population size	200	200	200	200
Selection operator	Random/stochastic uniform	Random/stochastic uniform	Random/stochastic uniform	Random/stochastic uniform
Elite	10	10	10	10
Crossover rate	0.5/0.8	0.5/0.8	0.5/0.8	0.5/0.8
Recombination	Wright's heuristic	Wright's heuristic	Wright's heuristic	Wright's heuristic
Mutation operator	Gaussian	Gaussian	Gaussian	Gaussian
Mutation standard deviation	0.3	0.3	0.3	0.3
Mutation shrink rate	1.0	1.0	1.0	1.0

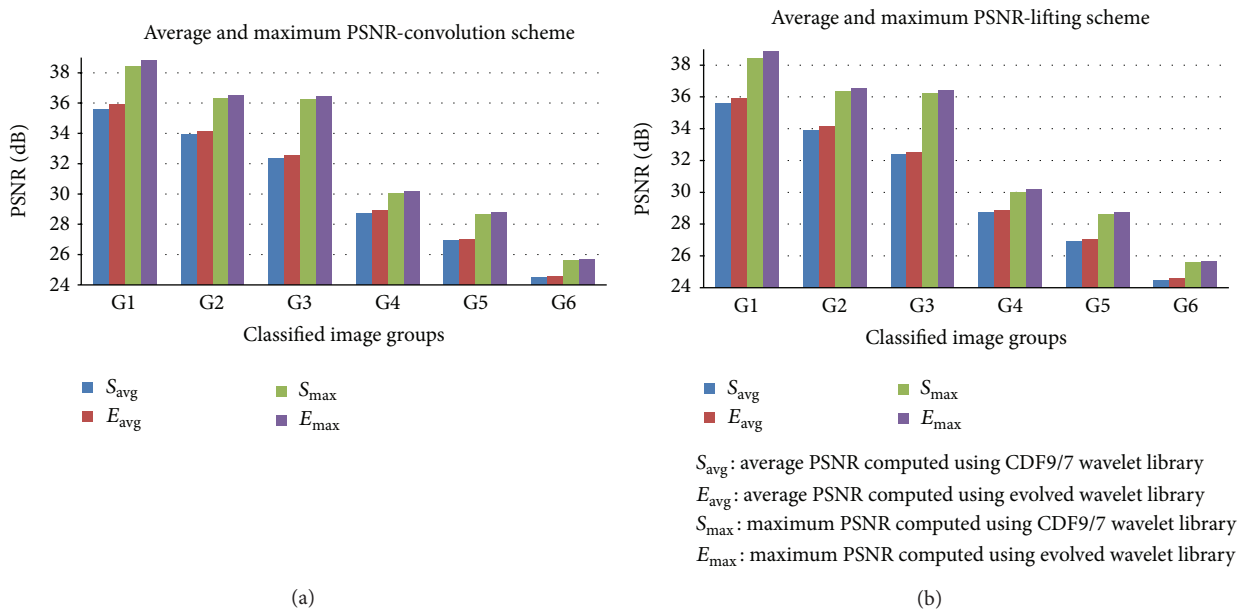


FIGURE 10: Comparison of quality metrics of the reconstructed image using CDF 9/7 and evolved filter coefficients; (a) average and maximum PSNR for convolution scheme; (b) average and maximum PSNR for lifting scheme.



Original image

Binary mask

Reconstruction of edge  
pixels using evolved  
local filterReconstruction of  
nonedge pixels using  
evolved global filter

FIGURE 11: Reconstruction of images using global and local filters.

enough to evolve 9 filter coefficients  $L1, L2, L3, L4, L5, H1, H2, H3,$  and  $H4$ .

*Evolving 2 Filter Coefficients.* In this method, the wavelets are considered to be biorthogonal. Hence only  $LR3$  and  $LR4$  will be evolved here and all other coefficients will be derived as follows [25]:

- (1)  $n = \sqrt{2}$ ;
- (2)  $LR2 = n/4 - LR4/2$ ;
- (3)  $LR1 = n/4 - LR3$ ;
- (4)  $L3 = (3n/4 - LR4)/(2 - 4n(LR4))$ ;
- (5)  $L4 = ((LR2^2 \times L3) + (LR3 \times LR2) - (LR4 \times LR1))/(2n/((LR3 \times LR2) - (LR2 \times LR1) - (LR4 \times LR1)))$ ;
- (6)  $L2 = n/4 - L4$ ;
- (7)  $L1 = -1 \times ((LR1 \times L2)/LR2)$ ;
- (8)  $L5 = 2 \times (L4 - L3 + L2 - L1)$ .

### 2.3.2. Lifting Scheme

*Evolving 5 Variables.* In this method all the 5 parameters ( $a, b, c, d, k$ ) are allowed to evolve randomly.

*Evolving 1 Variable.* In this method only  $k$  is allowed to evolve and all others are derived as follows [26]:

- (1)  $a = (1 - 2k)/4(k - 1)$ ;
- (2)  $b = -(k - 1)^2$ ;
- (3)  $c = 1/4k(k - 1)$ ;
- (4)  $d = k^3 - (7/4)k^2 + k$ .

## 3. Image Classification Based on Frequency Content

The DWT filter coefficients evolved for images with smooth regions might not suit well for edge and texture rich images. Also, it is not practical to construct the optimal wavelet

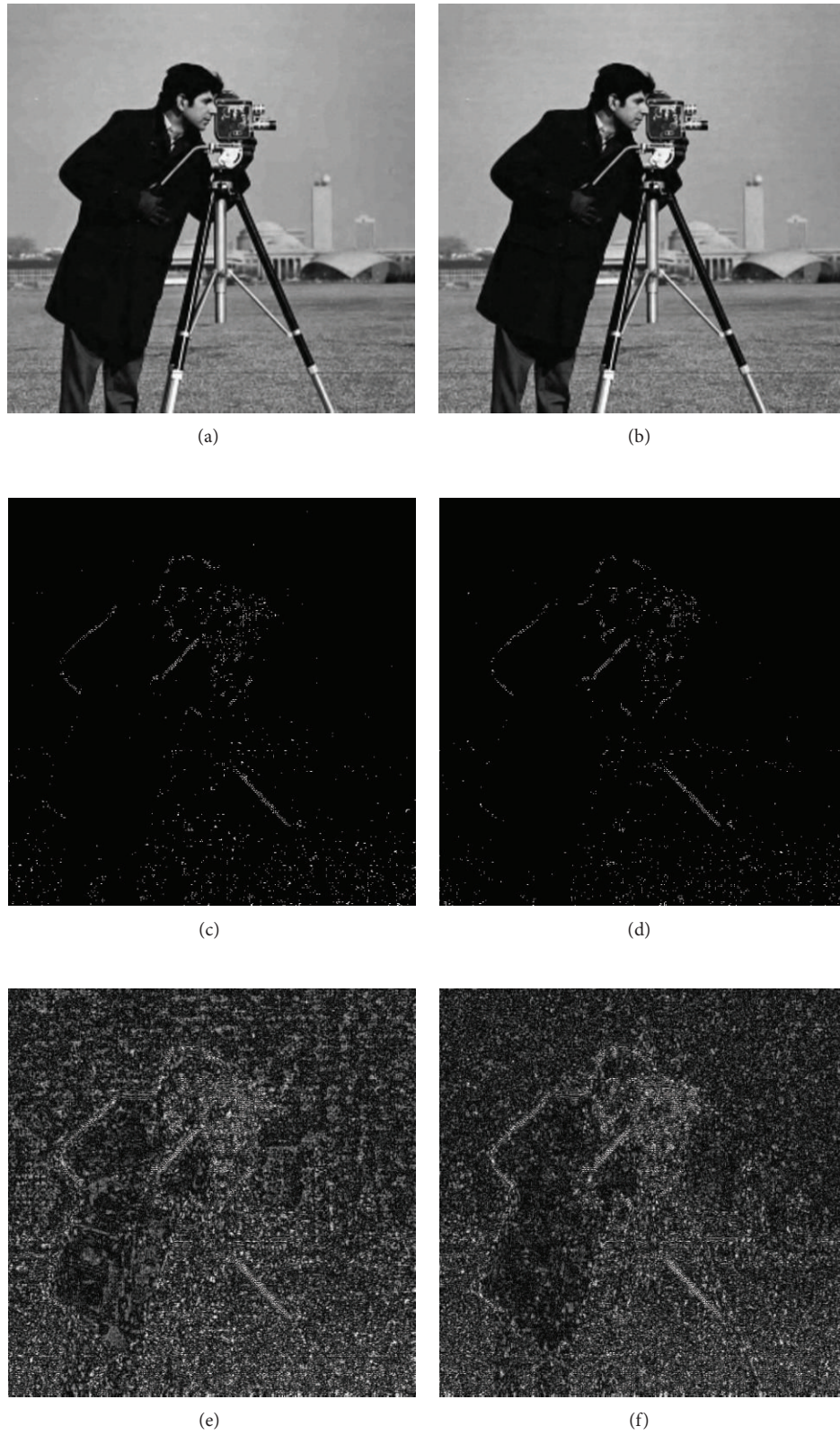


FIGURE 12: Comparison of evolved and CDF 9/7 filters. ((a), (b)) Reconstructed image using evolved and CDF 9/7 filter coefficients. ((c), (d)) Difference image using evolved and CDF 9/7 filter coefficients. ((e), (f)) Enhanced difference images for better visibility.

TABLE 4:  $F\_MEAN$  values for 50 test images.

Image	$F\_MEAN$
P1	10.5147
P2	5.5117
P3	16.1949
P4	18.9663
P5	17.2253
P6	3.9881
P7	15.0666
P8	15.5465
P9	22.0394
P10	17.5639
P11	20.7364
P12	14.3209
P13	16.4834
P14	26.1350
P15	18.7598
P16	11.6503
P17	34.2640
P18	18.4811
P19	29.4208
P20	25.5023
P21	15.2356
P22	6.8231
P23	16.8230
P24	14.8521
P25	7.5128
P26	10.8964
P27	11.5614
P28	13.0462
P29	16.7697
P30	8.3708
S1	28.6052
S2	25.3179
S3	30.5507
S4	68.5516
S5	32.3629
S6	30.7848
S7	50.1749
S8	27.4554
S9	48.5618
S10	45.0343
S11	51.6971
S12	44.3086
S13	47.0612
S14	32.6130
S15	34.4422
S16	38.5398
S17	42.2794
S18	46.8646
S19	42.5832
S20	44.8287

TABLE 5: Classification rule.

Complexity level	$F\_MEAN$
G1	0–9.999
G2	10–14.999
G3	15–19.999
G4	20–29.999
G5	30–44.999
G6	Above 45

for each image as an online process in spite of the best compression with the evolved filter coefficients. Hence all the test images are classified according to the complexity of the images (edges and textures) and optimal wavelets are evolved for each class to build a wavelet library offline. The quality of the DWT-based compression method for remote sensing images is effectively assessed using a gradient based approach by classifying image pixels according to the gradient magnitude and texture complexity thus proving the importance of the edges and textures in an image [27]. Hence we propose a systematic approach to find the edges and textures of the image by using the DWT itself. The high frequency subbands of transformed image will depict the edge and texture content in an image. Texture rich images will have more coefficients in the high frequency subbands as depicted in Figure 6 (3 cases are considered). This implies that the images can be classified by looking into the high frequency subbands. Thus the *Frequency\_Mean* (summation of absolute averages of all the high frequency subbands) in frequency domain of an image is taken as measure to classify the images.

**3.1. Test Images.** We have taken 50 images as shown in Figure 7 for each run and those 50 images are classified into six groups (G1, G2, ..., G6) and the wavelets are evolved separately for each group and all of them are classified according to *Frequency\_Mean* ( $F\_MEAN$ ).

**3.2. Calculation of  $F\_MEAN$ .** The  $F\_MEAN$  is calculated using the steps followed in the Figure 8 and the calculated  $F\_MEAN$  are shown in Table 4 for the considered 50 test images.

**3.3. Classification of Images.** The images are classified into one of the six groups (G1, G2, ..., G6) according to the  $F\_MEAN$  value and the corresponding classification rule is shown in Table 5. For more clarity, the classified images are categorized in Figure 9 according to their groups. Finally, a library is build offline by evolving wavelets for each group separately using RCGA with PSNR as the fitness function.

## 4. Experimental Results and Discussion

Initially the images are classified according to the edges and textures using the algorithm discussed in the Section 3.2. The initial classification step provides six groups of images with different texture details. The idea is to evolve wavelet

TABLE 6: Evolved wavelet libraries.

		Complexity level					
		G1	G2	G3	G4	G5	G6
9	Global	0.054629	0.012605	0.032392	0.029695	0.036221	0.03662
		-0.02452	-0.00053	-0.02269	-0.01927	-0.02057	-0.0227
		-0.06516	-0.07454	-0.07889	-0.07895	-0.07514	-0.08703
		0.334343	0.262207	0.279217	0.280296	0.309677	0.295225
		0.632333	0.69317	0.714242	0.707114	0.627418	0.683349
		0.143108	0.109115	0.032392	0.096019	0.147287	0.090116
		-0.09349	-0.04232	0.089464	-0.05766	-0.05681	-0.0556
		-0.54933	-0.56726	-0.05808	-0.5389	-0.58923	-0.53448
		0.99999	0.98762	-0.53136	0.99761	0.999996	0.997501
	Local	-0.14134	-0.11646	-0.09367	-0.09367	-0.16327	-0.09198
		-0.08997	-0.06448	-0.06013	-0.06013	-0.0764	-0.05909
		0.550634	0.555133	0.537672	0.537672	0.58579	0.53853
		0.97280	0.99986	0.93826	0.92341	0.999997	0.96312
		0.550634	0.555133	0.537672	0.537672	0.58579	0.53853
		-0.06832	-0.08022	-0.07966	-0.07966	-0.08858	-0.08863
		-0.32981	-0.2571	-0.28254	-0.28254	-0.31979	-0.30657
		0.626812	0.702748	0.703591	0.703591	0.621365	0.670795
		-0.32981	-0.2571	-0.28254	-0.28254	-0.31979	-0.30657
1	Global	1.293053	1.286464	1.263407	1.259675	1.273145	1.265296
	Local	1.290432	1.291972	1.276267	1.256783	1.27437	1.265312

TABLE 7: Comparison of other implementation methodologies.

Method	EA	Implementation	Seed	Image details	Max. PSNR value (dB)
Salvador et al. [7]	Simpler ES	Hardware: FPGA	Random Gaussian	Fingerprint image	36.9672
Salvador et al. [8]	Simpler ES	Hardware: FPGA	Random and CDF 9/7	FVC2000 fingerprint image	37.0146
Our Approach	Modified GA	Extrinsic (software)	CDF 9/7 with random noise	Versatile image set	38.8502
CDF 9/7	—	Extrinsic (software)	—	Versatile image set	38.4563

filter coefficients for each individual group both for near-edge and far-edge pixels in an image. Based on the output of the edge detection algorithm, a binary mask is created for the considered image and the binary mask separates the near-edge and far-edge pixels. The next step is to evolve wavelet filter coefficient for the near-edge pixels followed by far-edge pixels. The experiment is repeated for all the images which fall in the same group and the corresponding evolved filter coefficient is stored in the library. The experiment continues with the next group and concludes after evolving filter coefficients for all the six groups. Fifty GA runs are considered for both convolution scheme and lifting scheme and each GA run would consider one of the test images shown in the Figure 9. The GA configuration followed for conducting the experiment is given in Table 3.

Thus we have created an optimal wavelet library suitable for image compression for each class of images. For compressing an arbitrary image, its optimal wavelet filter coefficients need to be selected from the prestored library based on its  $F\_MEAN$  value which serves as an index for the selection of wavelets. We have evolved 9 filter coefficients for convolution scheme as the 2-variable evolver failed in most situations

to produce a better wavelet than the CDF 9/7. For lifting scheme single variable is evolved as the 5-variable evolver failed because of its NIL constraint situation. The evolved wavelet libraries for both global and local filter are shown in Table 6.

The comparison of the quality measures in convolution and lifting schemes are shown in Figure 10. Figure 11 shows the images reconstructed using global and local evolved filters and Figure 12 shows the comparison between images reconstructed using CDF 9/7 and evolved filters coefficients.

The ES based wavelet optimization algorithm discussed by Salvador et al. [7, 8] focused on hardware implementation choosing FPGA as the base fabric. The existing ES is modified to suit the hardware implementation with a hardware efficient mutation operator and it was tested for both floating point and fixed point arithmetic. Our focus is to improve the quality of reconstruction (PSNR) by evolving wavelet filter coefficients for image subgroups based on the texture and edges. Also, the same evolved filter coefficients may not suit for all image groups and hence we evolve different filters even if the improvement is marginal for the first level of decomposition. The improvement is best pronounced





as the decomposition level increases. Table 7 compares the optimization methodology and improvement in the results in terms of PSNR.

## 5. Conclusion and Future Work

Thus a lossy image compression method with improved performance compared to the CDF 9/7 based compression has been designed. The experimental results of Hybrid subband decomposer for the selected test images show significant improvement in the average PSNR and the maximum PSNR value for the reconstructed image subjected to quantization. Thus evolving wavelets for each group of images classified according to the  $F\_MEAN$  value is robust for performing lossy image compression. The evolved wavelets show an average improvement of 0.31 dB and a maximum improvement of 0.39 dB under convolution scheme. Under lifting scheme the evolved wavelets show an average improvement of 0.27 dB and a maximum improvement of 0.35 dB. Apart from using PSNR as the quality metric, the wavelets can be evolved by also considering SSIM and EC for the fitness measure to further improve the performance of compression. As extrinsic evolution of filter coefficients takes large amount of time, intrinsic evolution can be carried out by implementing an optimized light weight GA core on an FPGA platform so that filter coefficients can be evolved in lesser amount of time and hence make it suitable for adaptive systems.

## Appendix

### PSNR Values for Images

The 50 test images are grouped under six groups (G1, G2, ..., G6) and they are compressed and reconstructed using standard and evolved wavelet transform in the convolution scheme and their PSNR values are compared in Table 8. For lifting scheme the corresponding comparison is depicted in Table 9.

### Conflict of Interests

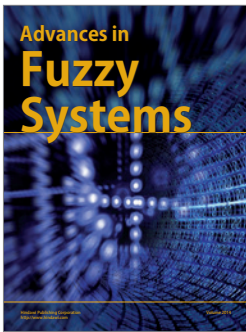
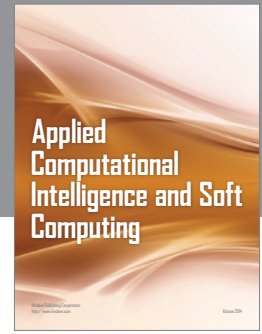
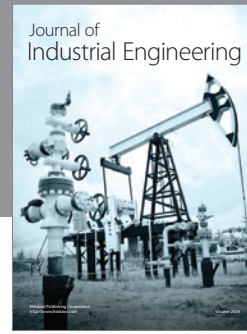
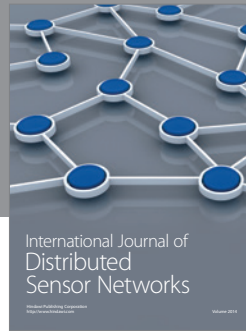
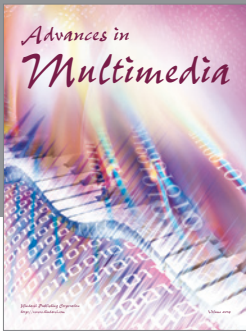
The authors declare that there is no conflict of interests regarding the publication of this paper.

## References

- [1] F. Moore, "A genetic algorithm for optimized reconstruction of quantized signals," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 105–111, 2005.
- [2] B. Babb, "Can evolved forward transforms do better than wavelet?" in *Proceedings of the 12th Annual Conference Genetic and Evolutionary Computation (GECCO '10)*, pp. 2043–2046, ACM, New York, NY, USA, July 2010.
- [3] B. J. Babb, F. W. Moore, and P. Marshall, "Evolved multi-resolution analysis transforms for improved image compression and reconstruction under quantization," in *Proceedings of the IEEE Symposium on Computational Intelligence in Image and Signal Processing (CIISP '07)*, pp. 202–207, Honolulu, Hawaii, USA, April 2007.
- [4] B. J. Babb, "Evolved transforms surpass the FBI wavelet for improved fingerprint compression and reconstruction," in *Proceedings of the 9th Annual Genetic and Evolutionary Computation Conference (GECCO '07)*, pp. 2603–2606, New York, NY, USA, July 2007.
- [5] U. Grasmann and R. Miikkulainen, "Effective image compression using evolved wavelets," in *Proceedings of the 7th annual conference on Genetic and evolutionary computation (GECCO '05)*, pp. 1961–1968, usa, June 2005.
- [6] U. Grasmann and R. Miikkulainen, "Evolving wavelets using a coevolutionary genetic algorithm and lifting," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '04)*, pp. 969–980, Seattle, Wash, USA, 2004.
- [7] R. Salvador, F. Moreno, T. Riesgo, and L. Sekanina, "Evolutionary approach to improve wavelet transforms for image compression in embedded systems," *Eurasip Journal on Advances in Signal Processing*, vol. 2011, Article ID 973806, 2011.
- [8] R. Salvador, F. Moreno, T. Riesgo, and L. Sekanina, "Evolutionary design and optimization of Wavelet Transforms for image compression in embedded systems," in *Proceedings of the NASA/ESA Conference on Adaptive Hardware and Systems (AHS '10)*, pp. 171–178, Anaheim, Calif, USA, June 2010.
- [9] M. R. Peterson and G. B. Lamont, "Improving image resolution with edge-targeted filter evolution," in *Proceedings of the IEEE Aerospace Conference (AC '08)*, pp. 1–14, Big Sky, Mont, USA, March 2008.
- [10] F. Moore, B. Babb, M. Peterson, and G. Lamont, "Evolved transforms improve image compression," SPIE Newsroom, 2009.
- [11] C. Miller, B. Babb, F. Moore, and M. Peterson, "Evolving improved transforms for reconstruction of quantized ultrasound images," in *Proceedings of the IEEE Workshop on Applications of Computer Vision (WACV '11)*, pp. 256–261, January 2011.
- [12] F. Moore, B. Babb, S. Aldridge, and M. R. Peterson, "Evolving matched forward and inverse transforms for improved lossy compression of images from Mars," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC '11)*, pp. 484–489, Anchorage, Alaska, USA, October 2011.
- [13] B. Babb and F. Moore, "The best fingerprint compression standard yet," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC '07)*, pp. 2911–2916, October 2007.
- [14] B. J. Babb, F. W. Moore, and M. R. Peterson, "Improved multi resolution analysis transforms for satellite image compression and reconstruction using evolution strategies," in *Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference (GECCO '09)*, pp. 2547–2552, 2009.
- [15] B. J. Babb, F. Moore, M. Peterson, and G. B. Lamont, "Improved satellite image compression and reconstruction via genetic Algorithms," in *Electro-Optical Remote Sensing, Photonic Technologies, and Applications II*, vol. 7114 of *Proceedings of SPIE*, pp. 1–8, 2008.
- [16] R. Salvador, F. Moreno, T. Riesgo, and L. Sekanina, "High level validation of an optimization algorithm for the implementation of adaptive Wavelet Transforms in FPGAs," in *Proceedings of the 13th Euromicro Conference on Digital System Design: Architectures, Methods and Tools (DSD '10)*, pp. 96–103, September 2010.
- [17] R. Salvador, A. Vidal, F. Moreno, T. Riesgo, and L. Sekanina, "Accelerating FPGA-based evolution of wavelet transform filters by optimized task scheduling," *Microprocessors and Microsystems*, vol. 36, no. 5, pp. 427–438, 2012.



- [18] K. T. Shanavaz and P. Mythili, "Faster techniques to evolve wavelet coefficients for better fingerprint image compression," *International Journal of Electronics*, vol. 100, no. 5, pp. 655–668, 2013.
- [19] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [20] M. R. Peterson, G. B. Lamont, and F. Moore, "Improved evolutionary search for image reconstruction transforms," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 2880–2887, July 2006.
- [21] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.
- [22] W. Sweldens, "The lifting scheme: a custom-design construction of biorthogonal wavelets," *Applied and Computational Harmonic Analysis*, vol. 3, no. 2, pp. 186–200, 1996.
- [23] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *The Journal of Fourier Analysis and Applications*, vol. 4, no. 3, pp. 247–269, 1998.
- [24] K. Kunaraj and R. Seshasayanan, "Leading one detectors and leading one position detectors—an evolutionary design methodology," *Canadian Journal of Electrical and Computer Engineering*, vol. 36, no. 3, Article ID 6704691, pp. 103–110, 2013.
- [25] B. Li, R.-H. Jiao, and Y.-C. Li, "Fast adaptive wavelet for remote sensing image compression," *Journal of Computer Science and Technology*, vol. 22, no. 5, pp. 770–778, 2007.
- [26] G. Yang, H. van de Wetering, and S. Zhang, "Optimization Design of Biorthogonal Wavelet Filter banks for Extending JPEG 2000 Standard Part-2," *Journal of Signal Processing Systems*, vol. 68, no. 2, pp. 247–259, 2011.
- [27] H. Jiang, K. Yang, T. Liu, and Y. Zhang, "Quality prediction of DWT-based compression for remote sensing image using multiscale and multilevel differences assessment metric," *Mathematical Problems in Engineering*, vol. 2014, Article ID 593213, 15 pages, 2014.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

