Eng. & Tech. Journal, Vol.28, No.13, 2010

Image Denoising Using Framelet Transform

Dr. Hadeel N. Abduallah* & Ali K. Nahar

Received on: 11/11/2008
Accepted on: 11/ 3/2010

Abstract

In many of the digital image processing applications, observed image is modeled to be corrupted by different types of noise that result in a noisy version. Hence image denoising is an important problem that aims to find an estimate version from noisy image that is as close to the original image as possible. In this paper, introduces firstly was applied method of computing one and two-dimensional framelet transform .The applying method reduces heavily processing time for decomposition of image keeping or overcoming the quality of reconstructed images. In addition, it cuts heavily the memory demands .Also, the inverse procedures of all the above transform for multi- dimensional cases verified. Secondly, many techniques are proposed for denoising of gray scale and color image. A new threshold method is proposed and compared with the other thresholding methods. For hard thresholding, PSNR gives (13.548) value while the PSNR was increased in the proposed soft thresholding, it gives (14.1734) PSNR value when the noise variance is (20). Some of the above denoising schemes are tested on *Peppers* image to find its effect on denoising application. The noisy version with SNR is equal to (11.9373 dB), the denoising image using WT with SNR is equal to (17.4661 dB), the denoising image using SWT with SNR is equal to (18.1459 dB), the denoising image using WPT with SNR is equal to (19.3640 dB), the denoising image using FT with SNR is equal to (21.9138 dB). Finally the denoising image for color image using FT with SNR is equal to (27.3443 dB).

Keywords: Fast Computing, Framelet Transform, Image Denoising, Filter Banks, Inverse Framelet Transform.

رفع الضوضاء عن الصوربأستخدام التحويل الاطاري

الخلاصة

في العديد مِنْ تطبيقاتَ مُعالجَة الصورةِ الرقميةِ، تصادف الصورةَ مُشَكّلَةُ التشويش بالأنواعِ المختلفةِ مِنْ الضوضاءِ التي تُؤدّي إلى نسخة مشوشة من الصورة. لِذلك رفع التشويش من الصورة من المشاكل المهمة التي تُهدّفُ لإيجاد نسخة تخمين مِنْ الصورةِ المشوشة كصورةِ محتملة قريبة من الصورةِ الأصليةِ. تُقدّمُ هذه المقالة أولاً، طريقة مقترحة لحساب النقل الإطاري ذو البعد الواحد وذو البعدين. الطريقة المُقتَرَحة تُخفضُ كثيراً من زمن معالجة الصورة المتحللة بالإضافة لذلك، تختصر كثيرا من الذاكرة المطلوبة. أيضاً تقدم التحويلات المعكوسة لكلّ الطرق في التحويل الاطاري في الإبعاد المتعددة. ثانياً، اقترحت العديد من التقنيات لرفع التشويش من الصور ذات اللون الرمادي والملون. واقترحت طريقة لتحديد حد العتبة وقورنت مع طرق تحديد العتبة الأخرى. بالنسبة لتحديد العتبة الصلب (Hard thresholding)، كانت نسبة الـــ(PSNR) هي (13.5483 dB) عندما كان توزيع التشويش هو (20). الـــ(PSNR) ازداد مع تحديد العتبة المعتدل (Soft thresholding)، فقد

***Electrical and Electronic Engineering Department, University of Technology/ Baghdad**

https://doi.org/10.30684/etj.28.13.3
2412-0758/University of Technology-Iraq, Baghdad, Iraq

أعطى قيمة (PSNR) مساوية إلى (14.1734 dB) بالوقت الذي كان فيه توزيع التشويش هو (20) .البعض مِنْ مخططاتِ رفع التشويش أعلاه مُجرّبة على صورةِ الفلفلِ لمعرفة تطبيقات رفع التشويش. فإذا كانت الحالة المشوشة للصورة لها قيمة نسبة الإشارة على الضوضاء (SNR) مساوية (١١.٩٣٧٣dB). تكون الصورة المخمنة العائدة من رفع التشويش باستخدام التحويل المويجي لها قيمة (SNR) مساوية إلى (١٧.٤٦٦١dB) ، والصورة المخمنة العائدة من رفع التشويش باستخدام ثابتة التحويل المويجي لها قيمة (SNR) مساوية إلى (١٨.١٤٥٩dB) ، والصورة المخمنة العائدة من رفع التشويش باستخدام حزمة التحويل المويجي قيمة (SNR) مساوية إلى (١٩.٣٦٤٠dB)، بينما اصبحت الصورة المخمنة العائدة من رفع التشويش باستخدام التحويل الإطاري لها قيمة (SNR) مساوية إلى (٢١.٩١٣٨dB)، أخيراً الصورة المخمنة العائدة من رفع التشويش باستخدام التحويل الإطاري للصورة الملونة كان قيمة (SNR) مساوية إلى (٢٧.٣٤٤٣ dB).

1. Introduction

Many real world images are contaminated by noise during their acquisition and/or transmission. In particular, multi-channel imaging is prone to quality degradation due to the imperfectness of the sensors often operating in different spectral ranges. In order to alleviate the influence of such disturbing artifacts on subsequent analysis procedures, denoising appears as a crucial initial step in multi-component image enhancement. In this context, attention has been paid to developing efficient denoising methods.

Though standard DWT is a powerful tool for analysis and processing of many real-world signals and images, it suffers from three major disadvantages, (1) Shift- sensitivity, (2) Poor directionality, and (3) Lack of phase information. These disadvantages severely restrict its scope for certain signal and image processing applications (e.g. edge detection,imageregistration/segmentation, motion estimation)[1-3] .

Other extensions of standard DWT such as Wavelet Packet Transform (WPT) and Stationary Wavelet Transform (SWT) reduce only the first disadvantage of shift- sensitivity but with the cost of very high redundancy and involved computation. Recent research suggests the possibility of exhausting all these disadvantages [1].Introduce the Double-Density Wavelet Transform (DDWT) as the tight-frame equivalent of Daubechies orthonormal wavelet transform; the wavelet filters are of minimal length and satisfy certain important polynomial properties in an oversampled framework. Because the DDWT, at each scale, has twice as many wavelets as the DWT, it achieves lower shift sensitivity than the DWT [2]

Two important facets need to be addressed, when resorting to the inherent frame redundancy: (1)Multiplicity: frame reconstructions arenotuniquein general,(2)Correlation: transformed coefficients (and especially those related to noise) are usually correlated, in contrast with the classical uncorrelateedness property of the components of a white noise after an orthogonal transform[4]. Although manycompression applications of wavelets use wavelet bases, other types of applications work better with redundant wavelet families, of which wavelet frames are the easiest to use. The redundant representation offered by wavelet frames has already been put to good use for signal denoising, and is currently explored for image compression [5] .

The framelet is an improvement upon the critically sampled DWT with important additional properties: (1) It employs one scaling function and two distinct wavelets, which are designed to be offset from one another by one half, (2) The double-density DWT is over-complete by a factor of two, and (3) It is nearly shift-invariant. In two dimensions, this transform outperforms the standard DWT in terms of denoising; however, there is room for improvement because not all of the wavelets are directional. That is, although the double-density DWT utilizes more wavelets, some lack a dominant spatial orientation, which prevents them from being able to isolate those directions

This paper describes new wavelet tight frames based on iterated oversampled FIR filter banks, first introduced in [6]. Selesnick et al. [6] introduce the double-density wavelet transform (DDWT) as the tight-frame equivalent of Daubechies' orthonormal wavelet transform; the wavelet filters are of minimal length and satisfy certain important polynomial properties in an oversampled framework. Because the DDWT, at each scale, has twice as many wavelets as the DWT, it achieves lower shift sensitivity than the DWT. New fast computation algorithms for computing discrete framelet transform have been described in this paper in a simple and easy to verify procedure based on iterated FIR filter bank that simplify computation complexity by using simple operations like matrix multiplication and addition.

2. Computation Method of Framelet Transform

The framelet transform is implemented on discrete-time signals using the over sampled analysis and synthesis filter bank shown in fig. (1). The analysis filter bank consists of three analysis filters- one low pass filter denoted by $h_0(n)$ and two distinct high pass filters denoted by $h_1(n)$ and $h_2(n)$. As the input signal $X(N)$ travels through the system, the analysis filter bank decomposes it into three sub bands, each of which is then down-sampled by 2. From this process $X_L(N/2)$, $X_{H1}(N/2)$ and $X_{H2}(N/2)$ are generated, which represent the low frequency (or coarse) subband, and the two high frequency (or detail) sub bands, respectively.

The up sampled signals are filtered by the corresponding synthesis low pass $h_0{}^*(n)$ and two high pass $h_1{}^*(n)$ and $h_2{}^*(n)$ filters and then added to reconstruct the original signal. Note that the filters in the synthesis stage, are not necessary the same as those in the analysis stage. For an orthogonal filter bank, $h_i{}^*(n)$ are just the time reversals of $h_i(n)$.

Wavelet frames, having the form described above, have twice as many wavelets than is necessary. Yet note that the filter bank illustrated in fig.(2) is oversampled by 3⁄2, not by 2. However, if the filter bank is iterated a single time on its lowpass branch (h_0), the total oversampling rate will be 7⁄4. For a three-stage filter bank, the oversampling rate will be 15⁄8. When this filter bank is iterated on its lowpass branch indefinitely, the total oversampling rate increases toward 2 .For computing fast discrete framelet transform consider the following transformation matrix for length-7[7, 8]:

$$W=\begin{bmatrix} h_0(0) & h_0(1) & h_0(2) & h_0(3) & h_0(4) & h_0(5) & h_0(6) & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & h_0(0) & h_0(0) & h_0(2) & h_0(3) & h_0(4) & h_0(5) & h_0(6) & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ h_0(2) & h_0(3) & h_0(4) & h_0(5) & h_0(6) & 0 & 0 & 0 & 0 & \cdots & h_0(0) & h_0(1) \\ h_1(0) & h_1(1) & h_1(2) & h_1(3) & h_1(4) & h_1(5) & h_1(6) & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & h_1(0) & h_1(1) & h_1(2) & h_1(3) & h_1(4) & h_1(5) & h_1(6) & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ h_1(2) & h_1(3) & h_1(4) & h_1(5) & h_1(6) & 0 & 0 & 0 & 0 & \cdots & h_1(0) & h_1(1) \\ h_2(0) & h_2(1) & h_2(2) & h_2(3) & h_2(4) & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & h_2(0) & h_2(1) & h_2(2) & h_2(3) & h_2(4) & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ h_2(2) & h_2(3) & h_2(4) & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & h_2(0) & h_2(1) \end{bmatrix}_{\frac{3N}{2}xN}$$

Here blank entries signify zeros, and for length-10 become:

$$W=\begin{bmatrix} h_0(0) & h_0(1) & h_0(2) & h_0(3) & h_0(4) & h_0(5) & h_0(6) & h_0(7) & h_0(8) & h_0(9) & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & h_0(0) & h_0(1) & h_0(2) & h_0(3) & h_0(4) & h_0(5) & h_0(6) & h_0(7) & h_0(8) & h_0(9) & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ h_0(2) & h_0(3) & h_0(4) & h_0(5) & h_0(6) & h_0(7) & h_0(8) & h_0(9) & 0 & 0 & 0 & 0 & \cdots & h_0(0) & h_0(1) \\ h_1(0) & h_1(1) & h_1(2) & h_1(3) & h_1(4) & h_1(5) & h_1(6) & h_1(7) & h_1(8) & h_1(9) & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & h_1(0) & h_1(1) & h_1(2) & h_1(3) & h_1(4) & h_1(5) & h_1(6) & h_1(7) & h_1(8) & h_1(9) & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ h_1(2) & h_1(3) & h_1(4) & h_1(5) & h_1(6) & h_1(7) & h_1(8) & h_1(9) & 0 & 0 & 0 & 0 & \cdots & h_1(0) & h_1(1) \\ h_2(0) & h_2(1) & h_2(2) & h_2(3) & h_2(4) & h_2(5) & h_2(6) & h_2(7) & h_2(8) & h_2(9) & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & h_2(0) & h_2(1) & h_2(2) & h_2(3) & h_2(4) & h_2(5) & h_2(6) & h_2(7) & h_2(8) & h_2(9) & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ h_2(2) & h_2(3) & h_2(4) & h_2(5) & h_2(6) & h_2(7) & h_2(8) & h_2(9) & 0 & 0 & 0 & 0 & \cdots & h_2(0) & h_2(1) \end{bmatrix}_{\frac{3N}{2}xN}$$

2.1. Computation Method of Framelet Transform for 1-D Signal

To compute a single level discrete framelet transform for 1-D signal the next steps should be followed:

1. Checking input dimensions: Input vector should be of length N, where N must be even and $N \geq$ length (analysis filters).
2. Construct a transformation matrix: Using transformation matrices given in Eqs. (1) and (2).
3. Transformation of input vector, which can be done by apply matrix multiplication to the *(3N/2)xN* constructed transformation matrix by the *Nx1* input vector.

$$Y = [W]_{\frac{3N}{2}\times N}[X]_{N\times 1} \quad \dots (3)$$

Where: Y is output vector.

$$X_L = Y\left[0:\frac{N}{2}-1\right]$$

$$X_{H1} = Y\left[\frac{N}{2}:N-1\right],$$

$$X_{H2} = Y\left[N:\frac{3N}{2}-1\right] \quad \dots (4)$$

2.2. Computation Method of Framelet Transform for 2-D Signal

There are two main types of methods for computing framelet transform for 2-D signal which are separable and non-separable algorithms. Separable methods simply work on each dimension in series. The typical approach is to process each of the rows in order and then process each column of the result. Non-separable methods work in both image dimensions at the same time.

Separable Method

A 2-D separable transform is equivalent to two 1-D transforms in series. To compute a single level discrete framelet transform for 2-D signal using separable method the next step should be followed:

1. Checking input dimensions: Input matrix should be of length NxN, where N must be even and N >= length (analysis filters).
2. Construct a transformation matrix: Using transformation matrices given in Eqs. (1) and (2).
3. Transformation of input rows by apply matrix multiplication to the *(3N/2)xN* constructed transformation matrix by the *NxN* input matrix.

$$Y = [W]_{\frac{3N}{2}\times N}[X]_{N\times N} \quad \dots .(5)$$

4. Transformation of input columns: can be done as follows:
   - **a.** Transpose the row transformed *(3N/2)xN* matrix resulting from step *3* .

$$[Y']_{N\times\frac{3N}{2}} = T\{\,[Y]_{3N/2*N}\,\}$$

.... (6)

b. Apply matrix multiplication to the *(3N/2)xN* constructed transformation matrix by the *NxN* column matrix.

$$YY = [W]_{\frac{3N}{2}\times N}[Y']_{N\times\frac{3N}{2}} \quad ....(7)$$

The final discrete framelet transformed matrix is equal to:

$$Y_o = [YY']_{\frac{3N}{2}\times\frac{3N}{2}} \quad ...\ (8)$$

Non- Separable Method

As simplifying computation complexity is one of the main concerns of this paper, and depending on the mathematical characteristics of the transformation matrices given in Eqs. (1) and (2), a fast orthogonal-based transform can be determined with more simplified computation procedure. Starting from the equivalence of two matrices theorem, the equivalence relation

$$Y = W\,X\,W^T \quad .....(9)$$

Is an orthogonal transformation and Y is orthogonally to X. So as the transformation matrices given in Eqs. (1) and (2) transposes are equal to their inverse matrices the orthogonal transformation of (4) can be used in computation of discrete framelet transform .

To compute a single level Orthogonal –based discrete framelet transform for 2-D signal the next steps should be followed:

1. Checking input dimensions: Input matrix should be of length *NxN*, where N must be even and $N >=$ length (analysis filters).
2. For an *NxN* matrix input 2-D signal *X*, construct a *(3N/2xN)* transformation matrix, W^T using transformation matrices given in Eqs.(1) and (2).
3. Apply Transformation by multiplying the transformation matrix by the input matrix by the transpose of the transformation matrix Y in Eqs.(9)

This multiplication of the three matrices result in the final discrete framelet transformed matrix.

2.3. Computation Method of Inverse Framelet Transform

1-D Inverse Framelet transform: To reconstruct the original signal from the discrete framelet transformed signal, inverse fast discrete framelet transform should be used. The inverse transformation matrix is the transpose of the transformation matrix as the transform is orthogonal.

To compute a single level 1-D Inverse discrete framelet transform, the following steps should be followed:

1. Let Y be the (3N/2)x1 framelet transformed vector.
2. Construct Nx(3N/2) reconstruction matrix, W^T, using transformation matrices given in Eqs.(1) and (2).
3. Reconstruction of input vector, which can be done by applying matrix multiplication to the Nx3N/2 reconstruction matrix, *T*, by the *3N/2x1* framelet transformed vector.

$$X = [W^T]_{N\times\frac{3N}{2}}[Y]_{\frac{3N}{2}\times 1} \quad ......\ (10)$$

2-D Inverse Framelet transform**:** To compute a single level 2-D Inverse discrete framelets transform using separable method the next steps should be followed:

1. Let Y_o be the (3N/2)x(3N/2) framelet transformed matrix.

2. Construct Nx3N/2 reconstruction matrix, W^T, using transformation matrices given in Eqs. (1) and (2).
3. Reconstruction columns: by applies matrix multiplication to the Nx(3N/2) reconstruction matrix by the (3N/2)x(3N/2) framelet transformed matrix.

$$YY = [T]_{N\times\frac{3N}{2}} [Y_o]_{\frac{3N}{2}\times\frac{3N}{2}} \quad \text{.... (11)}$$

4. Reconstruction rows: can be done as follows:

a. Transpose the column reconstructed matrix resulting from step 3.

$$Y = [YY']_{\frac{3N}{2}\times N} \quad \text{... (12)}$$

b. Apply matrix multiplication by multiplying the reconstruction matrix with the resultant transpose matrix.

$$X = [W^T]_{N\times\frac{3N}{2}} [Y]_{\frac{3N}{2}\times N} \quad \text{... (13)}$$

To compute a single level inverse framelet transform for 2-D signal using non-separable method the next steps should be followed:

1. Let Y_o be the (3N/2)x(3N/2) framelet transformed matrix.
2. Construct Nx3N/2 reconstruction matrix, W^T, using transformation matrices given in Eqs. (1) and (2).
3. Reconstruction of the input matrix by multiplying the reconstruction matrix by the input matrix by the transpose of the reconstruction matrix.

$$X = W^T Y_o W \quad \text{..... (14)}$$

3. A Denoising Algorithm for Gray image

By representing a noisy image in the framelet domain, large coefficients tend to be associated with the main structure of the image whereas smaller coefficients are mainly caused by noise. The main idea behind framelet domain denoising is that an image can be improved by modifying the coefficients with respect to their significance. These modification schemes fall into two categories: hard and soft thresholding.

Denoising using framelet thresholding refers to data reconstruction obtained by a nonlinear operation in wavelet transform domain. It has been introduced for reconstructing an unknown function from noisy observations.

The proposed method is a simple and easy to verify procedure based on iterated FIR filter bank that simplify computation complexity by using simple operations like matrix multiplication and addition. Fig. (3) gives a threshold-denoising model using framelet transform. The framelet shrinkage denoising of noisy signal *X*, in order to recover y as an estimate of original signal *x* is represented as a 4-step algorithm with as decomposition levels, FT as forward framelet transform and IFT as inverse framelet transform as shown in flowchart(1) in fig (4).

1. Apply steps of the framelet algorithm to get the low frequency (or coarse) coefficients, and the high frequency (or detail) coefficients, using 2D discrete framelet transform algorithm stated in section computation method of framelet transform for 2-D signal
2. Select the threshold type: either selection is of soft threshold type, hence soft-thresholding is means of

translating all coefficients towards zero by a certain amount defined by threshold value (*T*). The choice of threshold value is very crucial for a given signal for denoising., soft Thresholding function is given in the equation (15) [9]; or selection is of hard threshold type , the framelet coefficients (at each level) below threshold (*T*) are made zero and coefficients above threshold are not changed . Similarly, the hard thresholding function is given in the equation (16).

$$q_T^h(f) = \begin{Bmatrix} f & if\ |f| > T \\ 0 & if\ |f| \le T \end{Bmatrix} \quad ....(15)$$

$$q_T^s(f) = \begin{Bmatrix} f - T & if\ f > T \\ f + T & if\ \ f < T \\ 0 & if\ \ |f| \le T \end{Bmatrix} ....(16)$$

3. Apply thresholding to the framelet coefficients (leave the coarse coefficients)

4. Apply J steps of the inverse framelet algorithm to get the denoised signal,

4. Wavelet and Framelet Image Denosing

A comparison is drawn between image denoising using framelet transform with that using scalar wavelet transform and wavelet packet and stationary wavelet transform.

This comparison study is performed on a database which consists of two gray images. Table (1) compares the results SNR using both wavelet and framelet transform for a *Lena & peppers* images as shown in fig. (5) After using different SNR of noise on a database of gray images. From table (1), SNR of framelet more than it does in SRN for wavelet transform on the entire signal to noise ratio. For example (16.5289 db) SNR to noisy image becomes (21.4893 dB) in WT,(21.9067 dB)in PWT,(23.3835 dB) in SWT and (25.2202 dB) in FT, 'bestree2', 'wprec2'.While with various tree management utilities available that we discussed their theories previously there were WT,SWT,PW yet when performed with Matlab (wavelet toolbox) functions 'wavedec2' and 'waverec2', wavelet packet (WP) algorithm is implemented with functions 'wpdec2'. The SWT is implemented with functions 'swt2' and 'iswt2'. In fig. (6) the RMS error variations value of threshold schemes are plotted, note the RMS of the 2-D wavelet and wavelet packet is higher than that of the 2-D framelet , this is an evidence for what is mentioned above .

5. A Proposed Denoising Algorithm for Color Image

The same algorithm proposed so that retune image to three dimensions (i.e. color image), by repeat the step from 1 to 4 for red, blue and green levels as made to gray level so that become on the color image, whose dimension is (*N*N*3*), using vacant matrix with size three dimensions fill up the two dimensions matrix (y) in step 4 above the algorithm three times on followings as it is shown in flowchart (2) in fig (7)., compared between color and gray image as shown in figures. (10) and (9), with computed SNR for all examples where we saw denoising in color image was better than the image without color because some defects in color image were flee, and found the SNR had above 27.5dB while in the image without color had no overstep 22.5dB at Noisy image 15.9490dB.

6. A Computer Test

At a given level in the iterated filter bank, this separable extension produces nine 2D subbands. These subbands are illustrated in Fig (8). Since *L* is a low-pass filter *($h_0(n)$)* while both *H_1 and H_2* are high-pass filters *($h_1(n)$ and $h_2(n)$)*, the *H_2H_2, H_2H_1, H_1H_2, H_1H_1* subbands each have a frequency-domainsupport comparable to that of the *HH* subband in a DWT. A similar scheme creates the *H_1L, H_2L (LH_1, LH_2)* subband with the same frequency-domain support as the corresponding *HL (LH)* subband of the DWT, but with twice as many coefficients. A general computer program computing a single-level fast discrete framelet transform is written using Matlab V.7.0 for a general NxN 2-D signal (or gray image). As shown in Fig. (9a), the original "Peppers" image dimensions are 512x512 (NxN). Shown in Fig.(9b) noisy image with Gaussian noise , denoising image as shown in fig. (9c) .And in Fig.(10a) the original "Peppers" color image dimensions are 512x512x3 (NxNx3). Shown in fig. (10b) noisy image with Gaussian noise, denoising image as shown in Fig. (10c).

7. Conclusions

This paper presents a new framelets transform computation methods from 1D-2D that verify the potential benefits of framelets and gain a much improvement in terms of low computational complexity. A algorithm to image denoising using framelet transforms whether for color image or the image without color proved to provide good results where SNR value of = 22dB and with color value of = 27.5dB almost. Some methods of wavelet transform were compared SNR WT value of (17dB), SWT value of (18dB) and PWT value of (19dB) almost. More detail in figures (5), (6), (9) and (10) were introduced in this thesis. These figures were ranked according to their visual quality of performance.

References

[1] SHUKLA, P. D. "Complex Wavelet Transforms and Their Applications", A Dissertation Submitted of Signal Processing Division Department of Electronic and Electrical Engineering University of Strathclyde Scotland United Kingdom, October, 2003.

[2] Spaendonck, R. L. . Burrus, C. S. "A new directional, low-redundancy, complex wavelet transforms" Department of Electrical and Computer Engineering, Rice University, Houston, TX 77251-1892, USA,2001

[3] Fernandes ,C. A. . Spaendonck, R. L. . Burrus, C. S. "A directional, shift-Insensitive, low-redundancy, wavelet transforms" Department of Electrical and Computer Engineering, Rice University, Houston, TX 77251-1892, USA, 2001.

[4] Chaux, C. L. . Duval, A. B. . Pesquet, J. C. "A nonlinear stein based estimator for multichannel image denoising" IEEE Trans. on Inform, Vol.17, No. 14,December, 2007 .

[5] Daubechies, I. . Han, B. . Ron, A. . Shen, Z. " Framelets: mra-based constructions of wavelet frames " This work was supported by the US National Science Foundation under Grants DMS ٩٦٢٦٣١٩-، DMS-9872890, DBI-9983114. and

ANI-0085984, the U.S. Army Research Office, 27 November 2001.

[6] (Selesnick, I. W. Sendur, L. "Iterated oversampled filter banks and wavelet frames", in "Wavelet Applications in Signal and Image Processing VIII, San Diego, 2000," Proceedings of SPIE, Vol. 4119.

[7] Al-Taai, H. N. "A novel fast computing method for framelet coefficients", American Journal of Applied Sciences Vol.5 No.11pp. (1522-1527), 2008.

[8] N. Abdullah, A. N. Abdallah and W. Ali "A noval video denoising method based on 3D framelet transform" IEEE International Workshop on Digital info Tainmenet and Visualization (IWDTV2008) University Malaysia Terenggaun, Malaysia,2008.

[9] Jalobeanu,A. . Blanc, L. . . Zerubia, J. "Satellite image deconvolution using complex wavelet packets " Interaction homme-machine, connaissances Projet Ariana Rapport de recherche n 3955 ,June 2000 .

[10] H. N. Al-Taai "*A Novel Fast Computing Method for Framelet Coefficients*",American Journal of Applied Sciences Vol.5 No.11pp. (1522-1527), 2008 .

[11] Z.J.Saleh " Video Image Compression Based on Multiwavelets Transform " Ph.D. Thesis, Department of Electrical Engineering University of Baghdad, Iraq 2004.

Table (1) Comparison between denoising performance using various methods, *T=40*.

Name of image	Noisy image	WT	SWT	WPT	FT
	SNR (dB)	SNR (dB)	SNR (dB)	SNR (dB)	SNR (dB)
Lena	16.5289	21.4893	21.9067	23.3835	25.2202
Peppers	11.9373	17.4661	18.1459	19.3640	٢١.٩١٣٨

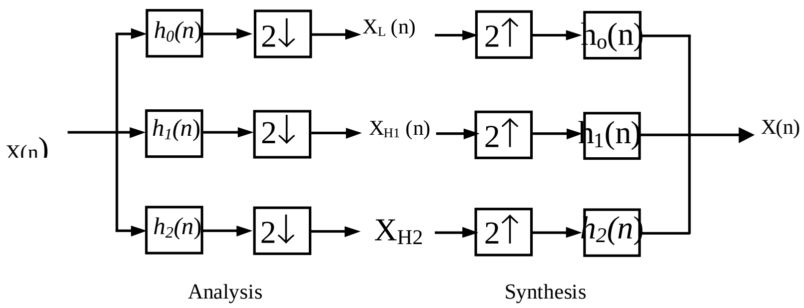


Figure (1) Analysis and Synthesis Stages of a 1-D Single Level Discrete Framelet Transform [10,11]

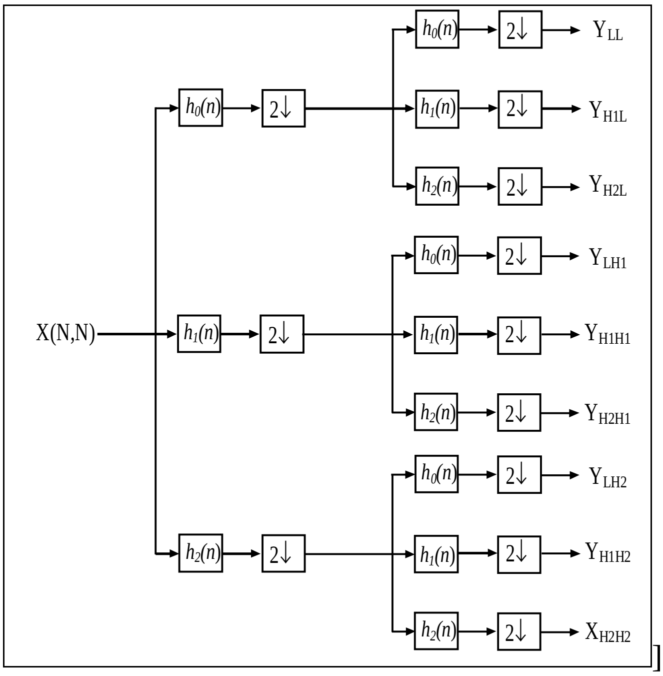


Figure (2): Analysis Stages of a 2-D Single Level Discrete Framelet Transform [10,11].

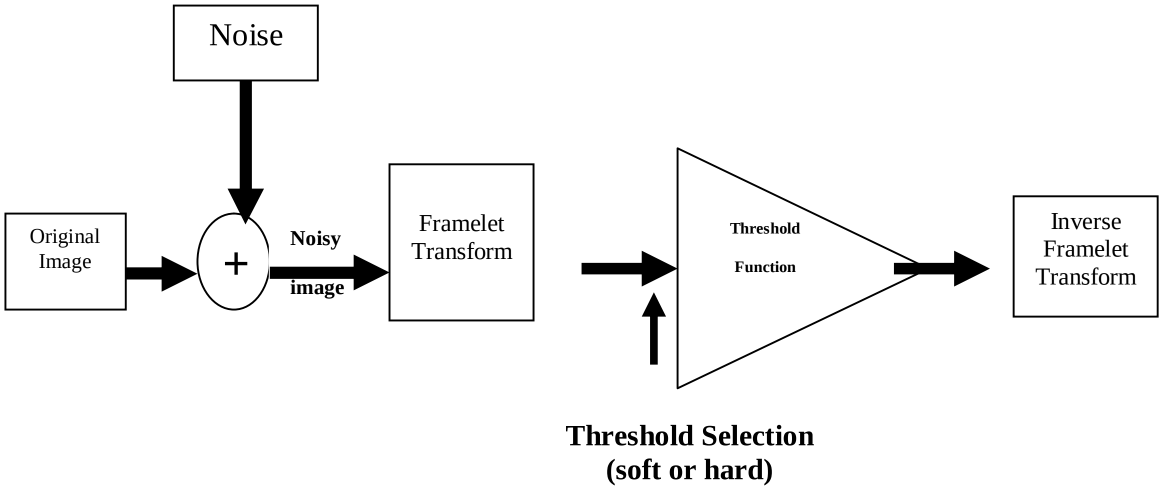


Figure (3) Image denoising structure using framelet transform

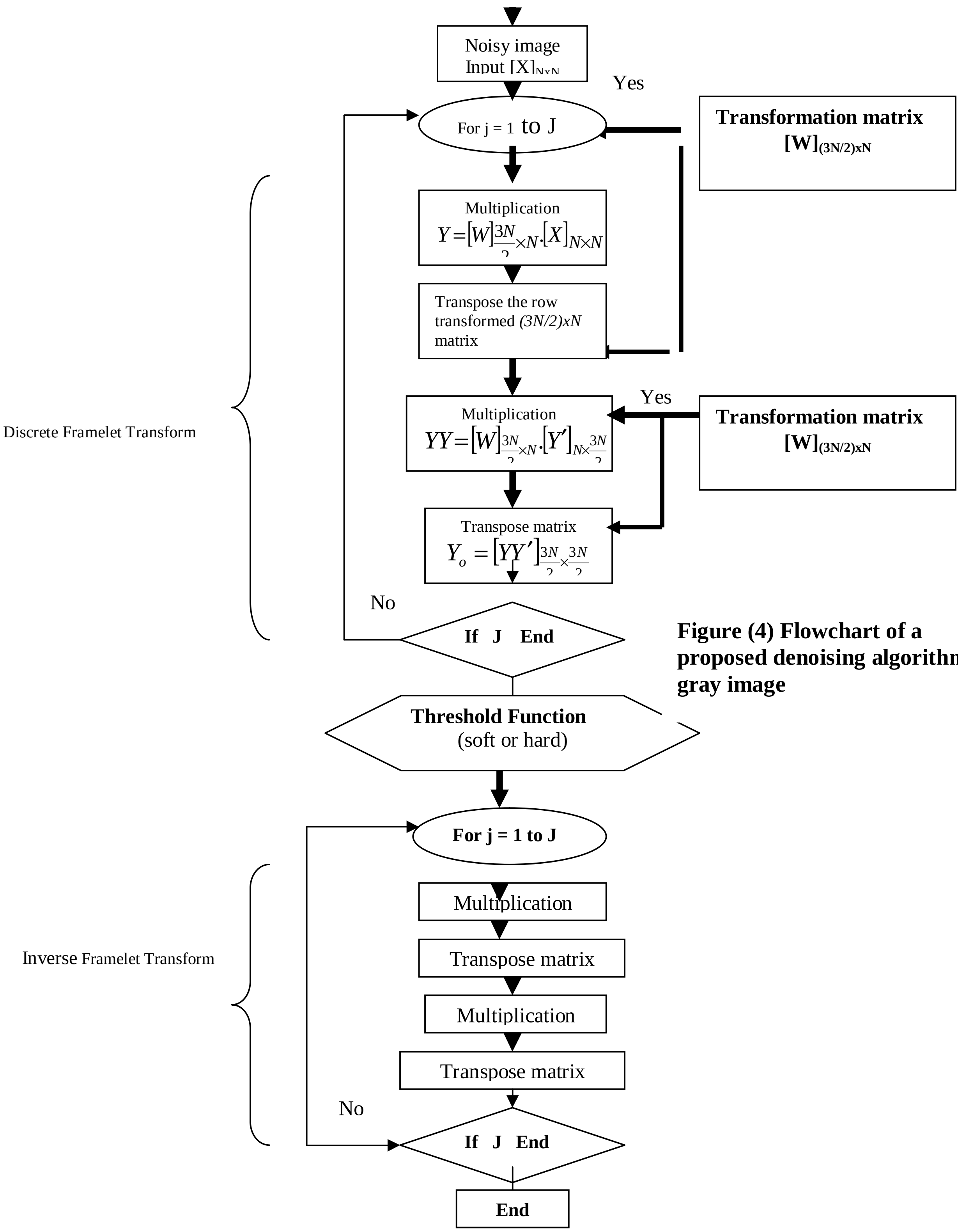


Figure (4) Flowchart of a proposed denoising algorithm for gray image

A) Original images

C) WT

E) WPT

Figure (5): SNR for various denoising methods for different images methods for denoising of 'Lenna' image with J=4, and wavelet type = db9.

B)**Noisy images**

E) SWT

F)FT

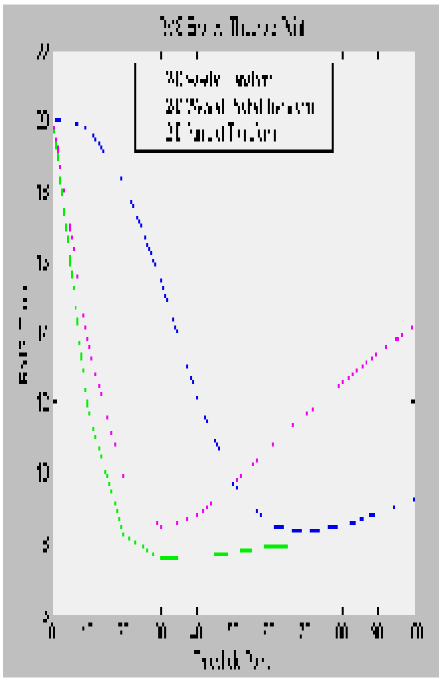


Figure (6): RMS Error variations vs. threshold plot for denoised image using discrete framelet transform and wavelet transform.

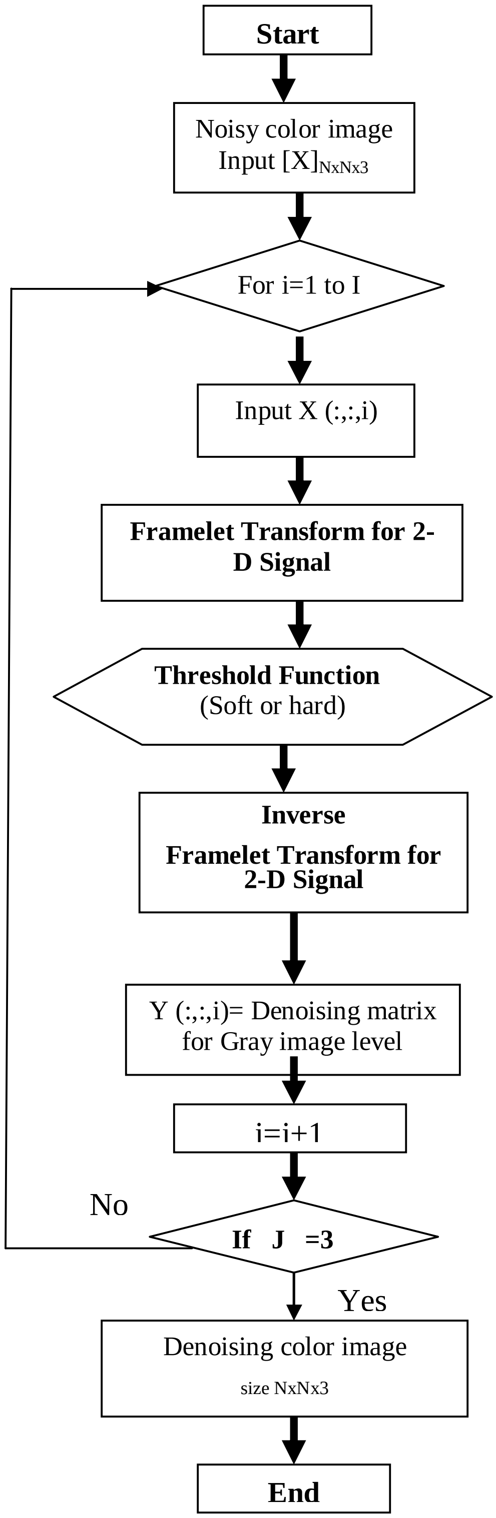


Figure (7) Flowchart of a proposed denoising algorithm for color image

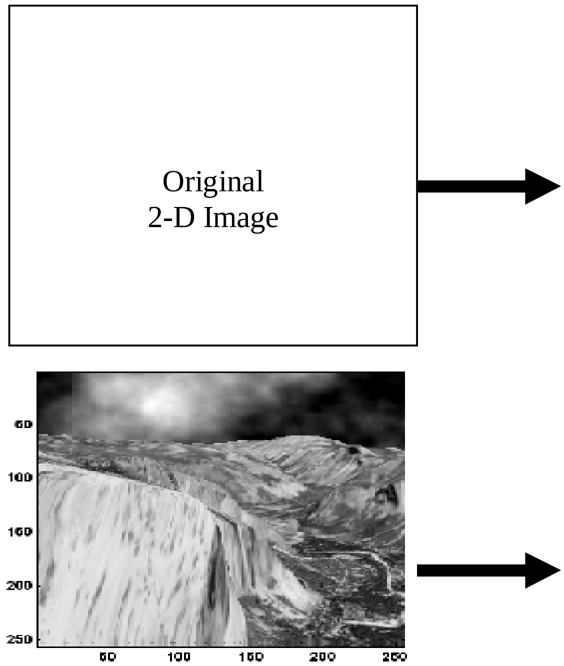
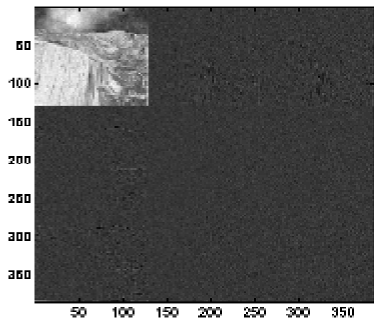
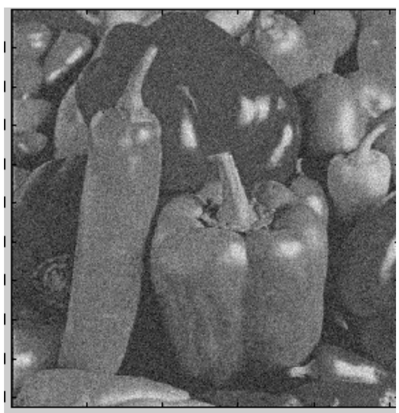


Figure (8): Image subbands after a single-level decomposition for framelet

(A)

LL	LH_2	LH_1
H_1L	H_1H_1	H_1H_2
H_2L	H_2H_1	H_2H_2

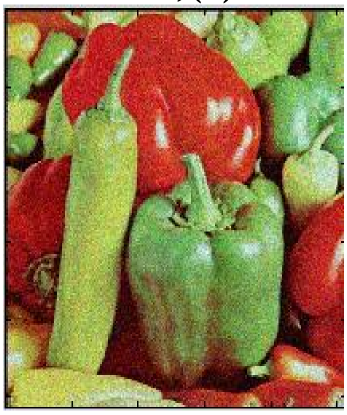




(B)

(C)

Figure(9): *Peppers* gray image (A) Original, (B) Noisy image with Gaussian noise , (C) Denoising image with J=4 and wavelet type =9db.



(A)

(B)

(C)

Figure (10): *Peppers* color image (A) Original, (B) Noisy image with Gaussian noise, (C) Denoising image