Color Image Denoising Using Local Multiwavelet Filtering

Dr. Waleed A. AL-Jauhar*, Matheel E. AL-Dargazli**

Received on: 25 /12 / 2003

Accepted on: 11 / 9 /2004

Abstract

This paper investigates the emerging notion of multiwavelets in the context of multirate filter banks, and applies a multiwavelet system to image denoising. Multiwavelets are of interest because their constituent filters can be simultaneously symmetric and orthogonal and because one can obtain higher orders of approximation for a given filter length. Local block filtering was a method for solving wavelet and multiwavelet transform. In this paper we add this method for image denoising application and comparing it with another methods of thresholding.

الخلاصة

هذا البحث يحقق الفكرة العامة لانبثاق متعدد المويجات في بيئة تعدد النسب لحدود المرشحات ويضيف نظام متعدد المويجات لرفع التشويش من الصورة. متعدد المويجات يسهم بشكل كبير بسبب إن مرشحاته الموحدة يمكنها أن تكون متناسقة وعمودية في الوقت نفسه ويمكن اضهار ترتيب أعلى من التقارب لطول المرشح المعطى. الترشيح باستخدام المجاميع المحلية هي طريقة للحل بالنقل المويجي وبمتعدد المويجات. في هذا البحث، أضيفت هذه الطريقة في تطبيق رفع التشويش من الصور ومقارنتها بطرق أخرى من حدود العتبة.

Keywords

Multiwavelets, image denoising, and local blocks.

1. Introduction

Multiwavelets systems with two or more scaling functions spanning the "low pass " space offer advantages of short support, symmetry, and orthogonality. While block filter banks issues of symmetric extension and filter design have not previously been addressed. Furthermore, the use of multiwavelet filters in a cascade algorithm leads to a novel pre-and post-processing method for block filter banks, based on the sampling/interpolation theory of wavelets [13]. Multiwavelets [6] used to solve different signal and image application like denoising and compression. This paper reviews different methods of denoising algorithms and comparing them with block processing with different type of algorithms that controlled on blocking scheme and size.

This paper is organized as follows: Section 2 reviews the definition and construction of continuous time multiwavelet system, and Section 3 describes the methods of denoising by threshold. In section 4 we introduce the new methods for block

*University of Baghdad / Baghdad, IRAQ

**Department of applied science, University of Technology, Baghdad, Iraq.

Construction of continuous time multiwavelet system and section 3 describes the methods of denoising by threshold. In section 4 we introduce the new methods for block processing multiwavelet transform. Finally, in Section 5. we compare the two above methods.

2. Multiwavelets

Multiwavelet bases of multiplicity 2 provide a multiresolution analysis usingthe multiscaling function and multiwavelet function. The perfect reconstruction multiwavelet filter bank is shown in Figure 1[8].

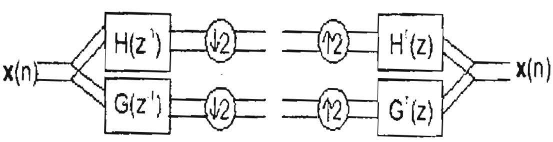


Figure 1: The perfect reconstruction multiwavelet filter bank (a) analysis section, (b) Synthesis section.

As in the scalar wavelet case, the theory of multiwavelets is based on the idea of multiresolution analysis (MRA) [1]. The difference is that multiwavelets have several scaling functions. The standard multiresolution has one scaling function $\varnothing(t)$ [8].

- The translates $\varnothing(t - k)$ are linearly independent and produce a basis of the subspace V_0.
- The dilates $\varnothing(2^j t-k)$ generate subspaces V_j, $j \in Z$.
- There is one wavelet $w(t)$. Its translates $w(t - k)$ produce a basis of the "detail" subspace W_0 to give V_1.

For multiwavelets, the notion of MRA is the same except that now a basis for V_0 is generated by translates of N scaling functions $\varnothing_1(t - k)$, $\varnothing_2(t - k)$, ..., $\varnothing_N(t - k)$. The vector $\varnothing(t) = [\varnothing_1(t), ..., \varnothing_N(t),]^T$, will satisfy a matrix dilation equation (analogous to the scalar case)

$$\Phi(t) = \sum_k C[k]\Phi(2t-k)$$

The coefficients C[k] are N by N matrices instead of scalars.

Associated with these scaling functions are N wavelets $w_1(t)$, ..., $w_N(t)$, satisfying the matrix wavelet equation

$$W(t) = \sum_k D[k]\Phi(2t-k)$$

Again, $W(t)= [w_1(t), ..., w_N(t)]^T$ is a vector and the D[k] are N by N matrices [10].

In practice multiscaling and wavelet functions often have multiplicity r=2. An important example was constructed by Geronimo, Hardin and Massopust [10], which we shall refer to as the GHM system. For the GHM multiscaling functions there are two scaling functions $\varnothing_1(t)$, $\varnothing_2(t)$ and the two wavelets $w_1(t)$, $w_2(t)$.

Another example of symmetric orthogonal multiwavelets with approximation

Order 2 is due to Chui and Lian (CL) [1]. Here both scaling functions are supported on [0,2], which is slightly longer than GHM. For the CL system, only three coefficients matrices are required, but it is less smooth than GHM ones.

Recently, Strela [1] suggested a way to construct biorthogonal multiscaling functions. Starting with the GHM system, one can take out one approximation order from the analysis part of the multifilter bank and transfer it to the synthesis part. GHM functions have two approximation orders, so the biorthogonal GHM (BiGHM) scaling functions have approximation order 3, and the dual ones have approximation order 1. Naturally, BiGHM scaling functions are smoother than dual ones.

Another biorthogonal multifilter bank BiHermite comes from Hermite cubic 'finite elements'. These cubics are supported on [0,2]. The Hermite multiscaling function satisfies a dilation equation with three coefficient matrices. The common feature of CL, BiGHM, and BiHermite systems is that one scaling function is symmetric and another is antisymmetric. Similar symmetry can be observed in the corresponding filter coefficients [1].

Before the operation of decomposition is applied to the input data, the preprocessing operation must be done. The aim of preprocessing is to associate the given scalar input signal of length N to a sequence of length-2 vectors $\{v_0,k\}$ in order to start the analysis algorithm. Here N is assumed to be a power of 2, and so is of even length. After the wavelet reconstruction (synthesis) step a postfilter is applied. Clearly, prefiltering, wavelet transform, inverse transform, and postfiltering should recover the input signal exactly if nothing else has been done. A different type of preprocessing was suggested such as repeated row (oversampling), matrix approximation (critical sampling) and so on [11].

A different type of prefilters and postfilters was used. The postfilter P that accompanies the prefilter Q satisfies PQ=I, where I is the identity matrix. Therefore, if we apply a prefilter, DMWT, inverse DMWT, a postfilter to any sequence, the output will be identical to the input. The commonly used prefilters are the identity prefilter, the Xia prefilter, the minimal matrix prefilter, the interpolation prefilter [11].

3. Thresholding Multiwavelets

Suppose that a signal of interest (f) has been corrupted by noise, so that we observe a signal (g) [2]:

$$g[n] = f[n] + \sigma z[n]$$

where z[n] is unit-variance, zero-mean Gaussian white noise. What is a robust method for recovering (f) from the samples g[n] as best as possible? Donoho and Johnstone [4,12] have proposed a solution via

Wavelet shrinkage (soft-thresholding) in the wavelet domain [3]. Wavelet shrinkage works as follows:

1. Apply the cascade algorithm to get the wavelet coeffcients corresponding to g[n].
2. Choose a threshold to the wavelet coefficients and apply it:

$$t_n = \sqrt{2\log(n)}\gamma\sigma / \sqrt{n}$$

3. Invert the cascade algorithm to get the denoised signal f‾ [n].

Donoho and Johnstone's algorithm [5] offers the advantages of smoothness and adaptation. Wavelet shrinkage is smooth in the sense that the denoised estimate (f) has a very high probability of being as smooth as the original signal f, in a variety of smoothness spaces. Wavelet shrinkage also achieves near minimum Root Mean Square-Error (RMSE) among possible of (f), measured over a wide range of smoothness classes. In these numerical senses, wavelet shrinkage is superior to other smoothing and denoising algorithms.

Heuristically, wavelet shrinkage has the advantage of not adding "bumps" or false oscillations in the process of removing noise, because of the local and smoothness preserving nature of the wavelet transform [7]. It is natural to attempt to use multiwavelets as the transform for a wavelet shrinkage approach to denoising, and Compare the results with scalar wavelet shrinkage.

In the oversampled scheme, the first row is multiplied by $\sqrt{2}$, to better match the first eigenvector of the GHM system. The critically sampled scheme uses the formulas below to obtain two input rows v1(n) and v2(n) from a single row of data. After reconstruction two output rows are deapproximated using the equations below to yield the output signal.

Although denoising by soft-thresholding is proven to be at least a smooth as the original function and free from spurious oscillations, there is a tradeoff between noise suppression and oversmoothing of image details. Another way of denoising by modifying the wavelet transform coefficients is called hard-thresholding and is expressed as:

$$\hat{X}k^{j} = T_h(G_k^j, Thrv) = \begin{cases} G_k^j, \left|G_k^j\right| > Thrv \\ 0, \left|G_k^j\right| \le Thrv \end{cases}$$

Hard thresholding yields better results than soft-thresholding in terms of RMSE, Signal to Noise Ratio (SNR), and for Peak SNR (PSNR). However it produces spurious oscillations.

4. Local Multiwavelet Filtering

In many of the transform based image denoising processing applications, distinct block transforms are used. That is, image is divided into $T \times U$ size distinct blocks, where both T and U are smaller than the image size, and $T \times U$ transform is performed over each block separately. The reason for this is not only to decrease the computational complexity, but also to exploit the local behavior of the image [9].

4.1. Image Denoising Using Local Distinct Block Processing

In distinct block operation, the noisy image is processed a block at a time, which means partitioning the noisy image in to $T \times U$ sections and some operation is performed on each distinct block individually to determine the values of the pixels in the corresponding block of the output image. So making use of multiwavelet transforms, the algorithm of image denoising using distinct block processing can be formulated as follows:

1. Divide the noisy image into $T \times U$ size distinct blocks.
2. Compute the multiwavelet transform coefficients $(G_k^j)^{(U\times T)} = MW^l g^{(U\times T)}$ of the distinct block $g^{(U\times T)}$.
3. Multiply the multiwavelet transform coefficients obtained in step (2) by the filter coefficients (a_k^j) to compute the estimated multiwavelet coefficients which corresponds to the output block:

$$(\hat{X}_k^j)^{(T\times U)} = a_k^j\, (G_k^j)^{(T\times U)}$$

4. Take the inverse multiwavelet transform to the estimated coefficients of step (3) to get the output (denoised) block:

$$\hat{x}^{(T\times U)} = (MW^l)^{-1}\, (\hat{X}_k^j)^{(T\times U)}$$

5. Repeat step 2 – 4 until all the distinct blocks are exhausted.
6. The denoised image is the concatenation of all the processed distinct blocks of step (4)

4.2. Image Denoising Using Local Sliding Neighborhood Processing In The Multiwavelet Domain

In some applications including image denoising and coding, data is parsed into non-overlapping blocks, denoising and coding processes are applied to each block separately. This independent processing of each block results in so called blocking effects. Blocking effects appear because the final samples of one block will, most likely, not match with the first samples of the next block. The modification made to the distinct block processing by overlapping the blocks with each other by extra rows and columns is not sufficient to remove the blocking effects totally. So, sliding neighborhood processing is developed to remove the blocking effects introduced by the previous algorithm. Image denoising using sliding neighborhood processing can be summarized in the following algorithm:

1.Compute the multiwavelet transform coefficients $(G_k^j)^{(t,u)} = MW^l g^{(t,u)}$ of the observed noisy image fragment $g^{(t,u)}$. $g^{(t,u)}$ is $T \times U$ frame from the observed noisy image including the pixels $(t,u),\ldots\ldots,(t+T-1,u+U-1)$.

2.Multiply the computed multiwavelet coefficients by the filter coefficients $\{a_k^j\}$:

$$(\hat{X}_k^j)^{(t,u)} = a_k^j \, (G_k^j)^{(t,u)}$$

3.Take the inverse wavelet transform to the estimated (filtered) wavelet coefficients of step (2) to get the output (denoised) frame.

$$\hat{x}^{(t,u)} = (MW^l)^{-1} (\hat{X}_k^j)^{(t,u)}$$

4.The central pixel of the fragment $\hat{x}^{(t,u)}$ is the estimated value of the image at pixel $(t+T/2, u+U/2)$.

5.Repeat the above steps by sliding the window over the image for $(t,u)=(0,0)$ to $(t,u)=(M,N)$.

4.3. Image Denoising Using Averaging Over Local Sliding Neighborhood Processing In The Multiwavelet Domain

In the previous algorithm, a sliding window over the observed data is used to estimate the central pixel value. Although this local transform approach within a sliding window is expected to adapt to the local characteristics of the image better than global transform domain approach, it may still result in similar artifacts as encountered in the traditional multiwavelet thresholding denoising. For obtaining a similar effects as translation-invariant denoising does, it is possible to keep the denoising results for every pixel in a window and estimate a pixel's values by averaging the corresponding pixel's outputs from denoising of several windowed fragments. Thus, the image denoising algorithm using averaging over local sliding neighborhood processing can be presented as follows:

1. Compute the multiwavelet transform coefficients $(G_k^j)^{(t,u)} = MW^L g^{(t,u)}$ of the observed noisy image fragment $g^{(t,u)}$. $g^{(t,u)}$ is $T \times U$ frame from the observed noisy image including the pixels $(t,u),\ldots\ldots,(t+T-1,u+U-1)$.

2. Multiply the obtained multiwavelet coefficients by the filter coefficients $\{a_k^j\}$:

$$(\hat{X}_k^j)^{(t,u)} = a_k^j \, (G_k^j)^{(t,u)}$$

3. Take the inverse multiwavelet transform to the estimated (filtered) multiwavelet coefficients of step (2) to get the output (denoised) frame.

$$\hat{x}^{(t,u)} = (MW^{L})^{-1}(\hat{X}_k^j)^{(t,u)}$$

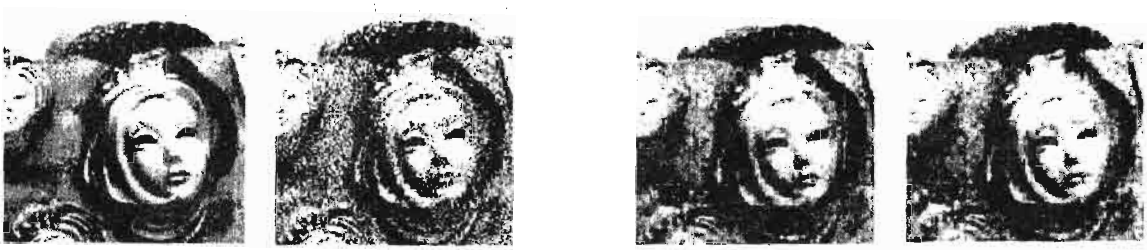
4. Store the denoising results for every pixel in the output frame that overlaps with another frame.
5. Repeat the above steps by sliding the window over the image for $(t,u) = (0,0)$ to $(t,u) = (M,N)$.
6. The denoised value for every pixel (r_1, r_2) in the output (reconstructed) image is the average of all the estimates obtained for that pixel from several fragments of the image enclosed by all

$W_{t,u}$ such that $(r_1, r_2) \in W_{t,u}$.

5. Numerical and Graphical Results

The algorithm is tested on 100 different color images. *Mask* image was one of them that we have see its results in this section with additive white Gaussian noise $\sigma = \{2\}$ to get the noisy version of this image.

Figure 2 shows the result of selecting different block size for overlapping it on all the image pixels. We conclude that whenever the block size is large, the result best.

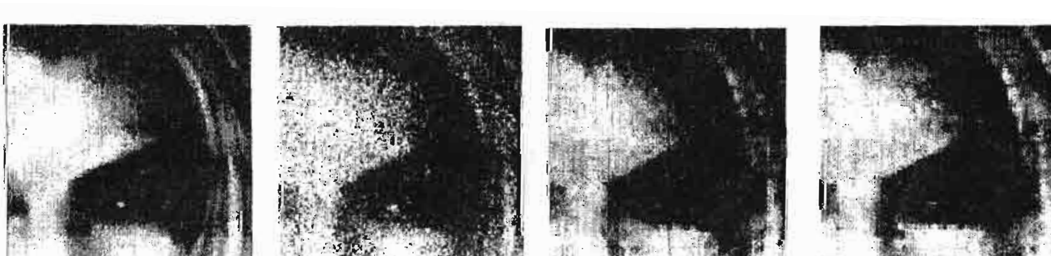


(a) (b) (c) (d)

(e)

Figure 2: Results of local distinct block processing algorithm using multiwavelet soft thresholding with CL basis functions and decomposition level=2, a) original mask image, b) Noisy version (RMSE= 87.4323), c) denoised image using LDBP algorithm with 16x16 block size (RMSE= 53.3372), d) denoised image using LDBP algorithm with 32x32 block size (RMSE= 51.6531), e) denoised image using LDBP algorithm with 64x64 block size (RMSE= 49.9743).

Ieal image was tested on different types of algorithms local sliding, minimum local sliding, and averaging local sliding. 32, 16, 8 was different sliding tested on those three algorithms. Averaging local sliding was the best results over all different blocking method. Figure 3 shows the images and its results. Table 1 present the objective measures of the previous results.



(a) (b) (c) (d)

(e)

Figure 3: Results of block processing algorithms using multiwavelet soft thresholding with CL basis functions and decomposition level=3, a) original ieal image, b) Noisy version (RMSE= 68.1306), c) denoised image using local sliding algorithm with 32x32 block size (RMSE= 26.5536), d) denoised image using local minimum over sliding algorithm with 16x16 block size (RMSE= 26.2173), e) denoised image using local averaging over sliding algorithm with 8x8 block size (RMSE= 26.1652).

The different thresholding algorithms scalar thresholding and local block processing thresholding (sliding, minimum, averaging) were compared to find the best one. We test these algorithms on Flower3 image that show it in Figure 4. From this comparison we see that local averaging over sliding neighborhood block processing was the best.

(a)

(b)

(c)

(d)

Figure 4: Results of block processing algorithms using multiwavelet soft thresholding with CL basis

functions and decomposition level=2, a) original ieal image, b) Noisy version (RMSE= 102.9753), c) denoised image using thresholding (RMSE= 57.4029), d) denoised image using local averaging over sliding algorithm with 8x8 block size (RMSE= 56.0702).

Denoising algorithm	Sliding size	RMSE
Sliding	32x32	26.5536
	16x16	27.8766
	8x8	30.1305
Minimum sliding	32x32	27.4053
	16x16	26.2173
	8x8	27.2672
Averaging sliding	32x32	26.4421
	16x16	26.1860
	8x8	26.1652

Table 1: Objective measures of ieal image with different algorithm results

7. Conclusions

Different types of thresholding algorithms were used for denoising images. Multiwavelet was used to translate these algorithms these methods was tested with different multiwavelet filters and decomposition levels of multiwavelet. Blocking effects was one of the most problems of blocking algorithms that solved by sliding with different types. In this paper, local sliding scheme of multiwavelet added to the denoising application and decrease the above problem. Also we can test this scheme on another application to see its measures.

References

1. V. Strela and A. Walden, "Orthogonal and biorthogonal multiwavelets for signal denoising and image compression", Dep. Of Math., Dartmouth College, 6188 Bradley Hall, Hanover, NH 03755, U.S.A, 1999.

2. T. R. Downie, B. W. Silverman, "The Discrete Multiple Wavelet Transform and Thresholding Methods", November 18, 1996.

*University of Baghdad / Baghdad, IRAQ
**Department of applied science, University of Technology, Baghdad, Iraq.

3.D. L. Donoho, "Denoising by soft-thresholding", IEEE Trans. on Inf. Theory, 41:6130{627, 1995.

4.D.L. Donoho and I.M. Johnstone, "Ideal spatial adaption via wavelet shrinkage", Biometrika, 81:425{455, 1994.

5. B. Jawerth and W. Sweldens, "An overview of wavelet based multiresolution analyses", SIAM Review, 36(3):377{412, 1994.

6. G. Strang and V. Strela, "Orthogonal multiwavelets with vanishing moments, Optical Engineering, 33:2104{2107, 1994.

7.G. Strang and V. Strela, "Short wavelets and matrix dilation equations", IEEE Trans. on Signal Processing, 43:108{115, 1995.

8. V. Strela, P.N. Heller, G. Strang, P. Topiwala, and C. Heil, "The application of multiwavelet filter banks to image processing", Technical report, Massachusets Institute of Technology, USA, 1995. Submitted to IEEE transform on Image Processing.

9.I. K. I. AL-Saleh, "Image denoising using wavelet transform", Master thesis in Baghdad Univ., Baghdad, Iraq.

10. J.-Ch. Pesquet, H. Karim, and H. Carfantan, "Time-Invariant orthonormal wavelet representations", IEEE Transactions on signal processing, Vol. 44, No. 8, August 1996.

11. X.-G. Xia, J. Geronimo, D. Hardin, and B. Suter, "On discrete multiwavelet transforms", IEEE Trans. on Signal Processing, 44:25{35, 1996.

12. D.L. Donoho, I.M. Johnstone, G Kerkyacharian, and D Picard, "Wavelet shrinkage: Asymptopia", Journal of the Royal Statistical Society, Series B, 57:301{369, 1995.

13. V.Strela, P.N.Heller, G.Strang, P.Topiwala and C.Heil, "Application of Multiwavelets to Signal Compression and Denoising", Dep. Of Math., Dartmouth College, 6188 Bradley Hall, Hanover, NH 03755, U.S.A, 1999.