# Denoising Based on the Adaptive Lifting

*Chang-Soo Lee, *Kyung-Yul Yoo

※This work is supported in part by the University Basic Research Program of the Ministry of Information and Communication


## Abstract

This paper introduces an adaptive wavelet transform based on the lifting scheme, which is applied to signal denoising. The wavelet representation using orthogonal wavelet bases has received widespread attention. Recently the lifting scheme has been developed for the construction of biorthogonal wavelets in the spatial domain. Wavelet transforms are performed through three stages: the first stage or Lazy wavelet splits the data into two subsets, even and odd, the second stage calculates the wavelet coefficients (highpass) as the failure to interpolate or predict the odd set using the even, and the third stage updates the even set using neighboring odd points (wavelet coefficients) to compute the scaling function coefficients (lowpass). In this paper, we adaptively find some of the prediction coefficients for better representation of signals and this customizes wavelet transforms to provide an efficient framework for denoising. Special care has been given to the boundaries, where we design a set of different prediction coefficients to reduce the prediction error.


## I. Introduction

The wavelet transform is an atomic decomposition that represents a real-valued continuous-time signal $x(t)$ in terms of shifted and dilated versions of a prototype bandpass wavelet function $\psi(t)$ and lowpass scaling function $\phi(t)$ [1]. For special choices of the wavelet and scaling function, the atoms

$$\psi_{j,k}(t) = 2^{j/2}\psi(2^j t-k), \quad j,k\in Z,\ j\le J \tag{1}$$

$$\phi_{J,k}(t) = 2^{J/2}\phi(2^J t-k) \tag{2}$$

form an orthonormal basis, and $x(t)$ could be decomposed into

$$x(t) = \sum_k c_k\phi_{J,k}(t) + \sum_{j=0}^{J}\sum_k d_{j,k}\psi_{j,k}(t) \tag{3}$$

with $d_{j,k}=\int x(t)\psi_{j,k}(t)dt$ and $c_k=\int x(t)\phi_{J,k}(t)\,dt$.

The wavelet coefficients $\{d_{j,k}\}$ and scaling coefficients $\{c_k\}$ comprise the wavelet transform. The set of scaling coefficients $\{c_k\}$ represents coarse signal information at scale $j=0$, whereas the set of wavelet coefficients $\{d_{j,k}\}$ represents detail information at scales $j=1,2,\cdot\cdot\cdot,J$. For a wavelet centered at time zero and frequency $f_0$, $d_{j,k}$ measures the content of the signal around the time $2^j k$ and frequency $2^{-j}f_0$ (equivalently, scale $j$). Wavelet ransforms of sampled signals could also be computed extremely efficiently using multirate filter bank structures. The forward discrete wavelet transform (DWT) could be implemented with a lowpass filter $h$ and highpass filter $g$. For the inverse DWT, a filter bank consists of a lowpass filter $\tilde{h}$ and a highpass filter $\tilde{g}$. For $\tilde{h}=h$ and $\tilde{g}=g$, it forms an orthogonal basis [1]. The representation of the data using wavelets coefficients offers an accurate approximation of $f$ by using only a few wavelet coefficients. It comes from the fact that the vanishing moments property of wavelets suppresses low-order polynomial signals in the highpass filter and we get a small fraction of wavelet coefficients [2, 3].

Recently, it has been shown that denoising, compression, and signal recovery methods based on wavelet coefficient shrinkage or wavelet series truncation not only have the asymptotic minimax performance characteristics, but also do not introduce excessive artifacts in the signal reconstruction. These exceptional performances come from the fact that wavelet bases are unconditional bases for many signal spaces. The unconditional nature of the wavelet basis is crucial to wavelet-domain processing, since it guarantees that the norm of the processed signal would not blow up even when some wavelet coefficients are discarded or reduced in magnitude [4].

The lifting scheme is a flexible tool for constructing wavelets without employing the Fourier transform and could therefore build wavelet bases over non-translation invariant domains such as bounded regions of $R^P$ or surfaces. The lifting algorithm is asymptotically twice as

* Dept. of Electrical Eng., Hanyang University

Manuscript Received : December 1, 1998

fast as the standard DWT algorithm and allows a fully in-place calculation of the wavelet transform without allocating auxiliary memory. The inverse wavelet transform could be found simply by undoing the operations of the forward transform. Also all wavelet transforms could be factored into the lifting steps with multiple predicts and updates [3, 5].

This paper concerns the application of lifting scheme to signal denoising. Thus, we introduce the adaptive lifting to customize the DWT and make the wavelet have the same correlation structure as the data. Through the adaptive lifting, the prediction error between the even and the odd set is to be minimized and then the prediction coefficients are computed adaptively. This paper is organized as follows. Section 2 provides the basics on the lifting scheme and discusses the design procedure for the predict and update stage. In Section 3 and 4, we review the wavelet denoising algorithm and apply the adaptive lifting to signal denoising. Finally Section 5 contains concluding remarks and future works.

## II. Lifting scheme

The lifting scheme could be used in situations where the Fourier transform is difficult to apply. Typical examples include [3];

① Wavelets on bounded domains: Construction of wavelets over an interval is required to transform finite length signals without introducing artifacts at the boundaries.

② Wavelets on curves and surfaces: To analyze the data on curves or surfaces or to solve equations on curves or surfaces, one needs wavelets intrinsically defined on these manifolds, independently of parameterization.

③ Weighted wavelets: Wavelets biorthogonal with respect to a weighted inner product are needed for diagonalization of differential operators and weighted approximation.

④ Wavelets and irregular sampling: Many real life problems require basis functions and transforms adapted to irregularly sampled data.

The lifting is a new method for constructing wavelets and it consists of following three stages.

① Split: Split the data into two smaller subsets $\lambda_{-1}$ and $\gamma_{-1}$. We refer to $\gamma_{-1}$ as the wavelet subset. The easiest possibility for the split is the Lazy wavelet.

② Predict: Make use of the $\lambda_{-1}$ subset to predict the $\gamma_{-1}$ subset based on the correlation present in the original data using a prediction operator $P$. It should be determined how many neighbors (vanishing moments) are used in the prediction.

$$\gamma_{-1} := \gamma_{-1} - P(\lambda_{-1}) \tag{4}$$

③ Update: Find a better $\lambda_{-1}$ so that a certain scalar quantity $Q()$ such as the mean is preserved, or

$$Q(\lambda_{-1}) = Q(\lambda_0) \tag{5}$$

Then construct an update operator $U$ and update $\lambda_{-1}$ as

$$\lambda_{-1} := \lambda_{-1} + U(\gamma_{-1}) \tag{6}$$

Three steps of the lifting are described on Figure 1. $\{\gamma_{-j}\}$ are computed by successively applying these three stages and represent the wavelet coefficients. $\{\lambda_{-j}\}$ are also lifted based on these wavelet coefficients and denote the scaling coefficients.

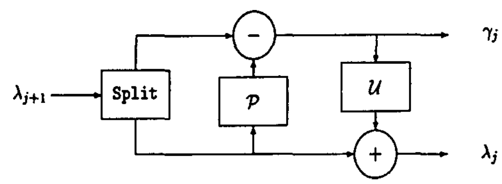



Figure 1. Structure of lifting scheme : Split, Predict, and Update.

### 2.1 Split

In this stage, we divide the signal into the even set $\{\lambda_{-j}\}$ and the odd set $\{\gamma_{-j}\}$. This mechanism is shown in Figure 2. At each level $j$, $\lambda_{-(j-1),2k}$ is set to $\lambda_{-j,k}$ and $\lambda_{-(j-1),2k+1}$ is set to $\gamma_{-j,k}$. In the simulation, we don't split the data physically, but the split and predict stages are combined into one function because of the in-place calculation [2, 3].

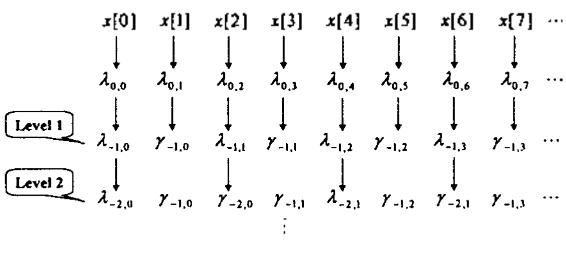



Figure 2. Schematic diagram explaining the split stage.

### 2.2 Predict

The prediction operator $P$ is based on the polynomial interpolation of order $N-1$ to find the predicted values. The higher the order of this operator, we could have the better approximation of the $\gamma$ coefficients based on the $\lambda$ coefficients. Through this prediction, we could suppress signals which resemble polynomials of order up to $N-1$ [6]. To prevent artifacts at the boundaries, we use a translation invariant lifting, which has to be modified near the left and right borders to construct bases whose supports remain inside the signal. For the cubic interpolation, we have four cases:

- Near the left boundary
  Case Ⅰ. 1 $\lambda$ on the left and 3 $\lambda$'s on the right
- Middle
  Case Ⅱ. 2 $\lambda$'s on the left and 2 $\lambda$'s on the right
- Near the right boundary
  Case Ⅲ. 3 $\lambda$'s on the left and 1 $\lambda$ on the right
  Case Ⅳ. 4 $\lambda$'s on the left and 0 $\lambda$'s on the right

To find the prediction coefficients, we first design a prediction operator $P$ so as to have better approximation of the signal of order up to $N-1$ [7]. For $N=4$, note that all polynomials of order up to $N-1$ would be suppressed if

$$\begin{bmatrix} -3^0 & -2^0 & -1^0 & 0^0 & 1^0 & 2^0 & 3^0 \\ -3^1 & -2^1 & -1^1 & 0^1 & 1^1 & 2^1 & 3^1 \\ -3^2 & -2^2 & -1^2 & 0^2 & 1^2 & 2^2 & 3^2 \\ -3^3 & -2^3 & -1^3 & 0^3 & 1^3 & 2^3 & 3^3 \end{bmatrix} \begin{bmatrix} -p_1 \\ 0 \\ -p_2 \\ 1 \\ -p_3 \\ 0 \\ -p_4 \end{bmatrix} = \mathbf{0}_{4\times 1} \quad (7)$$

Eq. (7) is simplified into

$$Vp = b, \quad (8)$$

where $V=\begin{bmatrix} -3^0 & -1^0 & 1^0 & 3^0 \\ -3^1 & -1^1 & 1^1 & 3^1 \\ -3^2 & -1^2 & 1^2 & 3^2 \\ -3^3 & -1^3 & 1^3 & 3^3 \end{bmatrix}$,

$$b=\begin{cases} [\ -2^0, \ -2^1, \ -2^2, \ -2^3\ ]^T, & \text{for case I} \\ [\ 1, \ 0, \ 0, \ 0\ ]^T, & \text{for case II} \\ [\ 2^0, \ 2^1, \ 2^2, \ 2^3\ ]^T, & \text{for case III'} \\ [\ 4^0, \ 4^1, \ 4^2, \ 4^3\ ]^T, & \text{for case IV} \end{cases}$$

and

$p \in (4\times 1)$ is the 4-point prediction coefficient matrix. Table 1 contains the filter coefficients for $N=4$, which are the solution of Eq. (8) for four different cases.

Table 1. Filter coefficients for each cases when $N=4$.

| Case | No. of $\lambda$'s on the left | No. of $\lambda$'s on the right | $p_1$ | $p_2$ | $p_3$ | $p_4$ |
|---|---|---|---|---|---|---|
| I | 1 | 3 | 0.3125 | 0.9375 | -0.3125 | 0.0625 |
| II | 2 | 2 | -0.0625 | 0.5625 | 0.5625 | -0.0625 |
| III | 3 | 1 | 0.0625 | -0.3125 | 0.9375 | 0.3125 |
| IV | 4 | 0 | -0.3125 | 1.3125 | -2.1875 | 2.1875 |

In the simulation, the filter coefficients matrix $p$ is constructed based on the information in Table 1.

$$p=\begin{bmatrix} 0.3125 & 0.9375 & -0.3125 & 0.0625 \\ -0.0625 & 0.5625 & 0.5625 & -0.0625 \\ 0.0625 & -0.3125 & 0.9375 & 0.3125 \\ -0.3125 & 1.3125 & -2.1875 & 2.1875 \end{bmatrix}$$

Figure 3 shows a 4-point prediction with the filter coefficients matrix with $p$. Arrows indicate the components used in the computation.

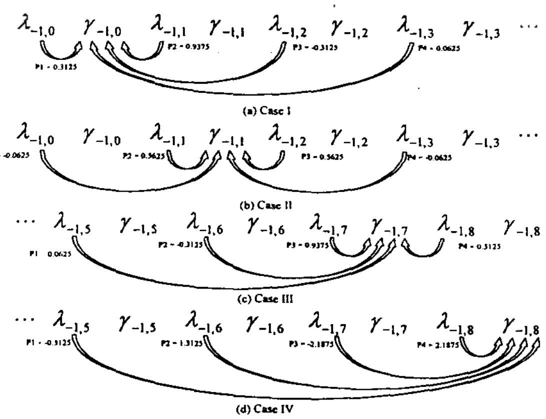


Figure 3. Elements used in the computation of 4-point prediction.

### 2.3 Update

The update stage preserves all the polynomials of order $N-1$. To find the update coefficients, we first initialize the integral-moment matrix $m^j \in (N\times L)$, where $L$ is the length of the signal. For $N=4$ at level $j=0$,

$$m^j = [m_0^j \ m_1^j \ m_2^j \ m_3^j \cdots] = \begin{bmatrix} 1 & 1 & 1 & 1 & \cdots \\ 0 & 1 & 2 & 3 & \cdots \\ 0 & 1 & 4 & 9 & \cdots \\ 0 & 1 & 8 & 27 & \cdots \end{bmatrix} \quad (9)$$

Once all the moments have been initialized, the moments corresponding to the $\lambda$'s have to be updated to preserve the average at every level. The idea is that each $\gamma$ coefficient would give back to the $\lambda$'s that were used to predict it as it received, and this amount is given by the prediction coefficients [6].

$$m^j_{2k+2\ell} \;:=\; m^j_{2k+2\ell} + p_{2,\ell+1} m^j_{2k+1}, \quad \ell = 0,1,2,3 \tag{10}$$

Assuming the signal length $L=16$, for $\ell = 0,1,2,3$,

(1) Case I : $m^j_{2\ell} \;:=\; m^j_{2\ell} + p_{1,\ell+1} m^j_1$

(2) Case II : ... $m^j_{2\ell+2} \;:=\; m^j_{2\ell+2} + p_{2,\ell+1} m^j_5$ ...

(3) Case III : $m^j_{2\ell+8} := m^j_{2\ell+8} + p_{3,\ell+1} m^j_{13}$

(4) Case IV : $m^j_{2\ell+8} := m^j_{2\ell+8} + p_{4,\ell+1} m^j_{15}$

Figure 4 shows the relationship between coefficients, $m^j_k$, on the moment update.

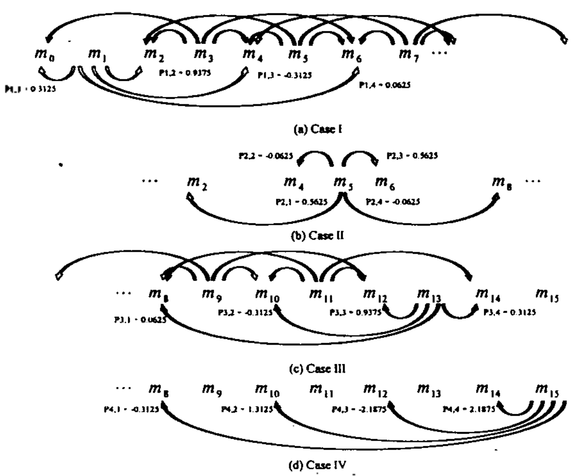



Figure 4. Moments update.

To find the update coefficients for every $\gamma$, we solve an equation:

$$[m^j_{2k-2}\ m^j_{2k}\ m^j_{2k+2}\ m^j_{2k+4}]u^j_k \;=\; {m^j_{2k+1}}^T \tag{11}$$

(1) Case I : $[m^j_0\ m^j_2\ m^j_4\ m^j_6]u^j_0 \;=\; {m^j_1}^T$

(2) Case II : ... $[m^j_2\ m^j_4\ m^j_6\ m^j_8]u^j_2 \;=\; {m^j_5}^T$ ...

(3) Case III : $[m^j_8\ m^j_{10}\ m^j_{12}\ m^j_{14}]u^j_6 \;=\; {m^j_{13}}^T$

(4) Case IV : $[m^j_8\ m^j_{10}\ m^j_{12}\ m^j_{14}]u^j_7 \;=\; {m^j_{15}}^T$

Finally, we update $\lambda$'s based on $\gamma$'s with the update coefficients. For each $\gamma$,

$$\lambda_{-j,k+l-1} := \lambda_{-j,k+l-1} \;+\; u^j_{l,k}\,\gamma_{-j,k}\,, \quad l=0,1,2,3 \tag{12}$$

(1) Case I : $\lambda_{-j,i} := \lambda_{-j,i} + u^j_{i,0}\,\gamma_{-j,0}\,,\ i=0,1,2,3$

(2) Case II :

... $\lambda_{-j,i+1} := \lambda_{-j,i+1} + u^j_{i,2}\,\gamma_{-j,2}\,,\ i=0,1,2,3$ ...

(3) Case III :

$\lambda_{-j,i+4} := \lambda_{-j,i+4} + u^j_{i,6}\,\gamma_{-j,6}\,,\ i=0,1,2,3$

(4) Case IV :

$\lambda_{-j,i+4} := \lambda_{-j,i+4} + u^j_{i,7}\,\gamma_{-j,7}\,,\ i=0,1,2,3$

### 2.4 Adaptive lifting

The prediction filter is designed so that it could eliminate polynomials up to order $N-1$, leaving only high-order polynomials. Therefore, the predict stage is equivalent to applying a highpass filter to the signal, whereas the update filter preserves low-order polynomials. However, if the signal doesn't have the polynomial structure, we would get large wavelet coefficients in magnitude. To optimize the approximation of signals with few wavelet coefficients, one could also construct adaptive wavelet bases with liftings that depend on the signal [8].

To represent the signal precisely, some ($M$) of the prediction points ($N$) are calculated to match the structure of the signal. In other words, we trade off them against the vanishing moments ($N-M$). $M$ points are used to adapt the prediction operator $P$ to the signal which has both polynomial and non-polynomial structure. At every scale $j$, we minimize the squared prediction error with the $N-M$ degrees of freedom to find the best prediction operator $P$ [7]. The vector of prediction error is then given by

$$\mathrm{e} \;=\; \mathrm{x_o} - \mathrm{X_e}\,p. \tag{13}$$

where $p_{\in(N\times 1)}$, $\mathrm{x}_{o\,\in(L\times 1)}$ is the odd $L$ samples to predict, and $X_{e\,\in(L\times N)}$ contains the neighboring even samples used in the predict stage. Then we solve the following minimization problem to find the prediction coefficients

$$\min_p \|\mathrm{x_o} - \mathrm{X_e}\,p\|^2 \quad \text{subject to Eq. (7)} \tag{14}$$

Note that the objective function consists of the sum of squared prediction errors $\mathrm{e^Te}$ with the $M$ polynomial constrains. This problem could be solved using the QR algorithm which could be found on [9].

## III. Denoising

Due to the concentrating ability of the wavelet transform, the DWT has been successively applied for noise removal. If a signal has its energy concentrated in a small number of wavelet dimensions, its coefficients

would be relatively large compared to any other signal or noise that has its energy spread over a large number of coefficients. This means that thresholding or shrinking the wavelet transform would remove the low amplitude noise or undesired signal in the wavelet domain, and the inverse DWT sill then retrieve the desired signal with little loss of detail [4, 10]. Because of good resolution in the both time and frequency domain, better noise separation with the wavelet transform other than with the spatial-domain or frequency-domain filtering is possible.
In [11], it has been shown that the denoising with the wavelet transform outperforms the FFT-based (frequency-domain) scheme when the input signal contains sharp edge. Also, it has been found that the wavelet denoising is effective in that although the noise is suppressed, edge features are retained without much loss. A drawback of the FFT is the fact that the edge information is spread across frequencies because of the basis functions not being localized in time or space, and hence the low-pass filtering results in the smearing of the edge. The denoising steps are given by

① compute the DWT $Y = Wy$, where $W$ is the wavelet transform matrix.

② perform thresholding in the wavelet domain, according hard thresholding

$$\hat{X} = T_h(Y, t) = \begin{cases} Y, & |Y| \ge t \\ 0, & |Y| < t \end{cases} \quad (15)$$

or according to soft thresholding

$$\hat{X} = T_s(Y, t) = \begin{cases} sgn(Y)(|Y| - t), & |Y| \ge t \\ 0, & |Y| < t \end{cases} \quad (16)$$

where $sgn(Y)$ is the sign value (-1 for negative and 1 for positive) of $Y$.

③ compute the inverse DWT $\hat{x} = W^{-1}\hat{X}$

For better denoising, we make use of the denoising based on soft thresholding the DWT coefficients [4]. The basis is adaptively changed according to the input signal to represent the signal effectively and it is applied to the DWT for denoising [7].

## IV. Simulation

Four signals, Blocks, Bumps, Heavy Sine, and Doppler [4] were tested for signal denoising. Each signal had 1024 samples and was corrupted by a white Gaussian noise with the standard deviation 1, where the standard deviation of the signals were varied from 5, 10, and 20.

In Table 2, the mean squared error (MSE) using Daubechies 8 (db8) as a basis and one with the adaptive lifting are compared. We set $N = \tilde{N} = 4$ and $M = 3$ for the construction of the lifting. Because of large wavelet and scaling coefficients, we reduced the vanishing moments to $N = \tilde{N} = 2$ and $M = 2$ at the boundaries. At only level 1, we found the optimized prediction coefficients using Eq. (14). The filter coefficients $p_2$, $p_3$, and $p_4$ were obtained through the algorithm in Section 2.2 and $p_1$ was adaptively optimized based on Eq. (14). According to the predict stage, the update coefficients were modified. We have calculated MSEs 100 times for each signal.

It is clearly seen from Table 2 that if the signal contains sharp transitions causing large amplitude wavelet coefficients, the adaptive lifting outperforms db8 because of the reduced vanishing moments. For Doppler, our implementation of the adaptive lifting has shown to be somehow unstable. Figure 5 shows a Blocks which is denoised through the adaptive lifting. In Figure 6, most of the noise is separated in the details D1 and D2. The MSEs in Table 2 show that the adaptive lifting performs nearly as good as db8.

Table 2. MSEs of signal denoising with different signals.

| SNR | Blocks | | | Bumps | | | Heavy Sine | | | Doppler | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | db8 | Lifting | Adaptive lifting | db8 | Lifting | Adaptive lifting | db8 | Lifting | Adaptive lifting | db8 | Lifting | Adaptive lifting |
| 5 | 0.5368 | 0.4617 | 0.4453 | 0.2394 | 0.2166 | 0.2198 | 0.0934 | 0.0869 | 0.0824 | 0.2013 | 0.2427 | 0.2134 |
| 10 | 0.4549 | 0.5157 | 0.4314 | 0.2527 | 0.3094 | 0.2911 | 0.1657 | 0.1464 | 0.1376 | 0.3332 | 0.3467 | 0.3442 |
| 20 | 0.5052 | 0.4934 | 0.4162 | 0.3835 | 0.4139 | 0.3857 | 0.2218 | 0.1910 | 0.1900 | 0.3839 | 0.5103 | 0.4234 |

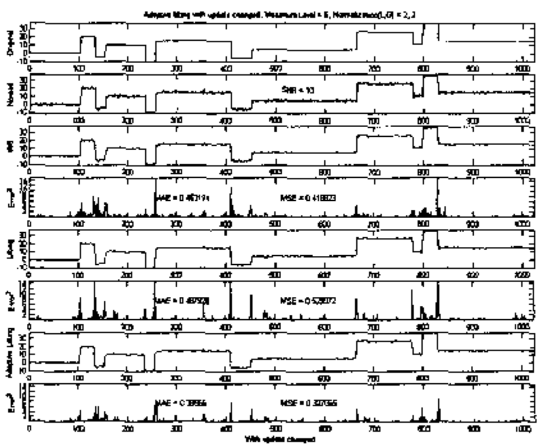


Figure 5. Blocks : Adaptive lifting with the update changed.

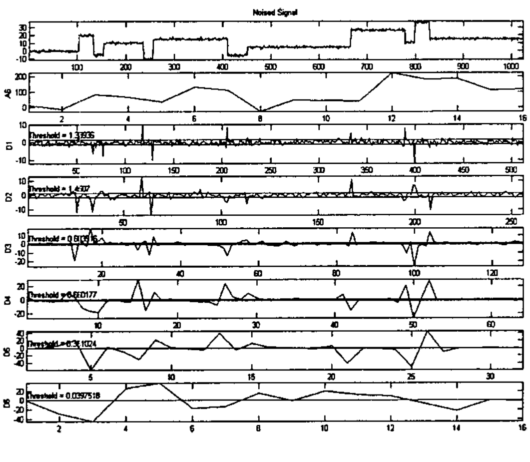



Figure 6. Approximation and details of the adaptive lifting for Blocks.

## V. Conclusions

In this paper, we have introduced the adaptive DWT based on the lifting with boundary cases considered. Using the lifting scheme it is particularly easy to adapt the DWT to the signals. The adaptive lifting could represent both smooth and edgy signal elements.

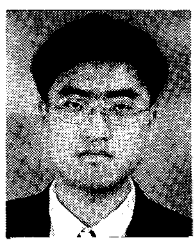
For good performance, the shorter wavelets are required in the neighborhood of singularities, whereas the longer wavelets with more vanishing moments could improve the approximation in regions where the signal is more regular [8]. In terms of results, the adaptive lifting becomes the Daubechies 6-like DWT when the signal has singularities. In contrast, its performance is close to Daubechies 8 when the signal is more regular, .

To achieve short support and high order approximation, one may use multiwavelets which offer simultaneous orthogonality, symmetry, and short support. A multiwavelet system could simultaneously provide perfect reconstruction while preserving length (orthogonality), good performance at the boundaries (via linear-phase symmetry), and a high order of approximation (vanishing moments). This would be a good candidate for the further research.

## References

1. Gilbert Strang and Truong Nguyen, *Wavelets and Filter Banks, Revised Edition*, Wellesley-Cambridge Press, 1997.
2. Wim Sweldens, "The lifting scheme : A custom-design construction of biorthogonal wavelets," *J. Appl. Comp. Harm. Anal.*, vol. 3, no. 2, pp. 186-200, 1996.
3. Wim Sweldens, "The lifting scheme : A new philosophy in biorthogonal wavelet constructions," *Wavelet Applications in Signal and Image Processing* III, pp. 68-79, Proc. SPIE 2569, 1995.
4. D. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inform. Theory*, vol. 41, pp. 613-627, May 1995.
5. Ingrid Daubechies and Wim Seldens, "Factoring wavelet transforms into lifting steps," Preprint, Bell Laboratories, Lucent Technologies, 1996.
6. Gabriel Fernandez and Senthil Periaswamy, "LIFTPACK : A Software package for wavelet transforms using lifting," *Wavelet Applications in Signal and Image Processing IV*, pp. 396-408, *Proc. SPIE* 2825, 1996
7. Roger L. Claypoole, Richard G. Baraniuk, and Robert D. Nowak, "Adaptive wavelet transforms via lifting," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing* '98, vol. , pp. 1513-1516, 1998.
8. Stephane Mallat, A *wavelet tour of signal processing*, Academic Press, 1998.
9. G. H. Golub and C. F. V. Loan, *Matrix Computations*, Johns Hopkins University Press, 1989
10. C. Sidney Burrus, Ramesh A. Gopinath, Haitao Guo, *Introduction to wavelets and wavelet transforms*, Prenctice Hall, 1998
11. Raghuveer M. Rao and Ajit S. Bopardikar, *Wavelet Transforms : Introduction to Theory and Applications*, Addison Wesley, 1998

▲Chang Soo Lee



Chang Soo Lee was born on March 27, 1972 in Suwon, Kyoungki, Korea. He received the B.S. degree in Electrical Engineering from Hanyang University, Ansan, Korea, in 1997. He is currently a graduate student for M.S. degree at Hanyang University, Seoul, Korea. His current research interests include digital signal processing, adaptive signal processing, and the wavelet transform and its application.

▲Kyung Yul Yoo

Kyung Yul Yoo received the B.E. degree in electrical engineering from Hanyang University, Korea, in 1982, the MS degree in electrical engineering and the MA degree in mathematics from the University of Arkansas, USA, in 1987 and 1989, respectively. He received the Ph.D. degree in electrical emgineering from the University of Missouri-Rolla, USA, in 1993.

From 1993 to 1995, he had been with the Electronic and Telecommunications Research Institute (ETRI), Daejeon, Korea, as a senior member of technical staff, and involved in the development of digital communication systems. Since 1995, he has been with the department of electrical engineering, Hanyang University, Korea, as an assistant professor. His current interest include statistical signal processing, wavelet, and adaptive filtering application.