Proceedings of the 28th IEEE
EMBS Annual International Conference
New York City, USA, Aug 30-Sept 3, 2006

Wavelet Compression of Multichannel ECG Data by Enhanced Set Partitioning in Hierarchical Trees Algorithm

Ershad Sharifahmadian, *Student Member, IEEE*
Department of Biomedical Engineering, Faculty of Engineering
Shahed University, Tehran, Iran
Postal Code:1991816543
E-mail:esahmadian@netscape.net

Abstract -The set partitioning in hierarchical trees (SPIHT) algorithm is very effective and computationally simple technique for image and signal compression. Here i modified the algorithm which provides even better performance than the SPIHT algorithm. The enhanced set partitioning in hierarchical trees (ESPIHT) algorithm has performance faster than the SPIHT algorithm. In addition, the proposed algorithm reduces the number of bits in a bit stream which is stored or transmitted. I applied it to compression of multichannel ECG data. Also, i presented a specific procedure based on the modified algorithm for more efficient compression of multichannel ECG data. This method employed on selected records from the MIT-BIH arrhythmia database. According to experiments, the proposed method attained the significant results regarding compression of multichannel ECG data. Furthermore, in order to compress one signal which is stored for a long time, the proposed multichannel compression method can be utilized efficiently.

I. Introduction

The large amount of data produced by ECG monitoring and recording facilities needs to be trasmitted, stored and analyzed in reasonable amounts of time. Therefore, it is required to compress data and use automated analysis [1], [2]. Various data reduction techniques for single channel ECG's have been presented in the literature [3]–[6].

In this paper, first, the enhanced set partitioning in hierarchical trees (ESPIHT) algorithm which is a modified version of the SPIHT algorithm, is presented. Also, i proposed a specific procedure based on the modified algorithm for compression of multichannel ECG data, in order to extend the single channel ESPIHT to the multichannel ESPIHT.

In the following section, the wavelet transform is briefly reviewed. In section III, the one-dimensional ESPIHT algorithm is explained but modifications can be considered for compression in other dimensions. Furthermore, the specific procedure for multichannel ECG compression based on the ESPIHT algorithm, is given in section IV. In section V, the modified algorithm is tested using selected ECG records from the MIT-BIH arrhythmia database and the simulation results are presented, moreover, the results of the multichannel ESPIHT implementation are presented. Section VI concludes the paper.

II. Wavelet Transform

Wavelet transform is widely used in many medical applications especially in signal detection, classification, denoising, and compression. It is often used in compression methods because of its energy compaction ability. A signal can be represented by scaling and translating a short wave called wavelet, ψ . Discrete coefficients which describe the scaling and translations are called wavelet coefficients. The wavelet transform can be represented as a dyadic filter bank with *M*-level as seen in Fig.1.

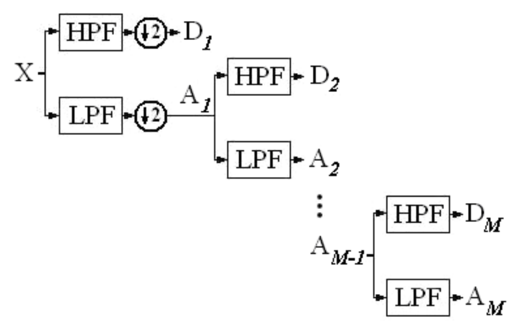

Fig.1. Dyadic filter bank with *M*-level

For a one-dimensional signal, each wavelet transform level can be realized using a pair of filters as shown in Fig.1. In the decomposition stage, the original signal X is passed through a low pass filter (LPF) and a high pass filter (HPF). Then , the two filtered outputs are down-sampled by two to obtain two subbands, approximation part *(A)* and detail part *(D)*, respectively. For most physiological signals, the signal energy is concentrated in the lower frequency subbands, thus this representation gives energy compaction. Many of the wavelet coefficients, especially in the higher frequency subbands, are either zero or close to zero. By coding only the larger coefficients, many bits are already omitted without loosing important data. ECG signals are non-stationary, every beat cycle consists mainly of QRS complex, P, T, S-T segment, baseline, and so on. The first three components are most useful in clinical diagnosis. As in [7], most of information about ECG signals concentrates in the low frequency subbands. Whereas the major energy of the QRS complex concentrates between 5-15 Hz, and the major energy of the P and T waves concentrate below 5 Hz, it is required the number of layers of the wavelet decomposition, and the filter bank are selected appropriately. The number of layers should be at least four for appropriate compression. The biorthogonal 9/7 tap filters are suitable which were applied successfully for ECG compression [3] and image compression [8].

III. Proposed Algorithm

After the M-level wavelet decomposition, it is obtained $M+1$ coefficient subbands. A one level wavelet decomposition produces two sets of coefficients: approximation coefficients (A_1), and detail coefficients (D_1). The length of A_1 and D_1 which are the same together, is computed according to the expression:

$$L_A = \begin{cases} \dfrac{L_X}{2} & \text{if } L_X \text{ is even} \\ \dfrac{L_X+1}{2} & \text{if } L_X \text{ is odd} \end{cases} \tag{1}$$

where L_X is the length of an input signal, and L_A is the length of the approximation part (which is equal to the length of the detail part).

When the wavelet coefficients are produced, they will be compressed using the ESPIHT algorithm. The principles of this algorithm are similar to the SPIHT algorithm which are partial ordering of the wavelet coefficients by magnitude with a set partitioning sorting algorithm, ordered bit plane transmission and exploitation of self-similarity across different subbands. Since most of the energy of a signal is concentrated in the low frequency subbands, the wavelet coefficients are arranged in hierarchies, called *temporal orientation trees*, with roots in the lowest frequency subband, branching consecutively into higher frequency subbands at the same temporal orientation [3]. In the tree structure, $O(i)$ is the set of offspring of a tree node defined by location (i). Each node either has no offspring or two offspring. The nodes at the highest frequency subband have no offspring by definition of the temporal orientation tree. The set of all descendants of a node defined by location (i), is named a type A set and denoted by $D(i)$. The set of all descendants excluding offspring of a node defined by location (i) is named a type B set and denoted by $L(i)=D(i)-O(i)$. During compression, significant information of the wavelet coefficients is placed in three ordered categories: 1) the category of insignificant positions (CIP) which contains individual coefficients that have magnitude smaller than a given threshold, 2) the category of significant positions (CSP) which contains positions found to have magnitude larger than (or equal to) a given threshold, and 3) the category of insignificant sets (CIS) that contains the sets of the wavelet coefficients which are defined by tree structures, and they had been found to have magnitude smaller than a given threshold, the sets exclude the coefficient corresponding to the tree or all subtree roots, and have at least two children. The CIS is defined as a matrix with two columns. The wavelet coefficients are encoded in several passes. In each pass only the coefficients with magnitudes exceeding a certain threshold are encoded. The first step in the ESPIHT algorithm is to determine the length of each wavelet subband according to equation (1). Then, the threshold of each wavelet subband is determined. The threshold of the jth wavelet subband T_j is given by the following

$$T_j = 2^{\left\lfloor \log_2 (Max_{i\in jth\ subband}\ |c_i|) \right\rfloor} \tag{2}$$

where c_i is the wavelet coefficient at location (i).

Among threshold of the wavelet subbands which were determined, the largest threshold is selected for the initial operational threshold. Afterwards the initial operational threshold is compared with the threshold of each subband. This comparison is performed from the highest frequency subband to the lowest frequency subband, and it will be stopped if the threshold of a subband equals the initial operational threshold for the first time. Then, the coding algorithm determines the position number of the last coefficient which belongs to the wavelet subband where comparison was performed up to that subband. This position number is designated as a limit which restricts evaluation which is performed to determine the significance of $D(i)$ (or $L(i)$), and this limit is named ***frontier***. For demonstration of this operation, i use an example (Example I). A four level wavelet decomposition of an input signal of length 64 produces the 64 wavelet coefficients distributed among the subbands, as shown in Fig.2. The length of each subband is determined according to equation (1).

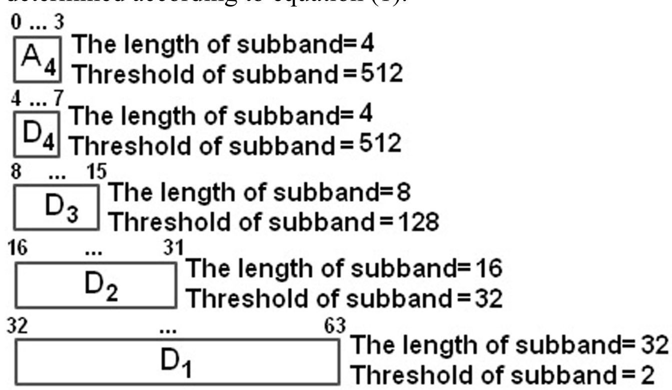

Fig.2. The length and threshold of the wavelet subbands in Example I

In this example, the length and threshold of each subband are shown. According to Fig.2, the initial operational threshold for this example is 512. In order to determine the frontier, the initial operational threshold is compared with 2, 32, 128, and 512, respectively. For the first time, the threshold of the fourth detail subband will be the same with the initial operational threshold, and the position number of the last coefficient which belongs to the fourth detail subband is designated as the frontier, i.e. the frontier is set as 7. Therefore, evaluation of the coefficients which belong to $D(i)$ (or $L(i)$) is performed up to the coefficient at location (7) in the first level of compression. In other words, evaluation which conduces to find this matter that $D(i)$ (or $L(i)$) is significant or insignificant, is performed up to the coefficient at location (7).

The category of significant positions (CSP) is set as empty. The category of insignificant positions (CIP) and the category of insignificant sets (CIS) are set according to the following rule:

L_A =length(approximation subband);
if L_A is even
add the locations {(0) to (L_A -1)} into the CIP;
add the locations {($\frac{L_A}{2}$) to (L_A -1)},with type A into the CIS;
else
add the locations {(0) to (L_A -2)} into the CIP;
add the locations {($\frac{L_A-1}{2}$) to (L_A -1)},with type A into the CIS;
end

It ought to be noted that insignificant sets are placed in the first column of the CIS. Then, the ESPIHT begins checking individual coefficients in the CIP (***sorting pass of the CIP*** is started). A coefficient will be significant if it is larger than (or equal to) the current operational threshold. If a coefficient is significant, a one is sent followed by a sign bit and its position number is moved to the CSP otherwise a zero is sent and its position number is preserved until the next level of compression. If the sign is positive, the sign bit is set as 1 and if the sign is negative, the sign bit is set as 0. After checking coefficients in the CIP, the coding algorithm begins checking the CIS (***sorting pass of the CIS*** is started). For a type A set ($D(i)$) in the first column of the CIS, when it is significant, a one is sent. Then, its two offspring are evaluated like a coefficient in the CIP. Afterwards the position number ($4i+3$) is checked, if this position number is smaller than (or equal to) the position number of the last wavelet coefficient in the highest frequency subband, the type A set ($D(i)$) is moved to the second column of the CIS and changed to type B ($L(i)$). If the position number ($4i+3$) is larger than the position number of the last wavelet coefficient, the position number ($4i$) is checked, if this position number is larger than the position number of the last wavelet coefficient in the highest frequency subband, the type A set is removed from the first column of the CIS. If the position number ($4i$) is smaller than the position number of the last wavelet coefficient, coefficients in the positions ($4i$), ($4i+1$), and ($4i+2$), are checked separately.
If $D(i)$ is insignificant, a zero is sent, the type A set is removed from the first column of the CIS, its position number and its type are preserved until the next level of compression. When all sets in the first column of the CIS were checked, the coding algorithm starts to test type B sets (or set) in the second column of the CIS.
Before checking each $L(i)$, the two offspring which are direct children of a wavelet coefficient at location (i), are considered again. In other words, these offspring which are direct descendants of a tree node at location (i) and they were previously evaluated, are very important. Evaluation of the significance of $L(i)$ means that checking all descendants in the subtrees corresponding to locations ($2i$) and ($2i+1$), in order to find a coefficient which is larger than (or equal to) the current operational threshold. This process is performed for each $L(i)$, and it is time-consuming. Therefore, if these two offspring are smaller than the current operational threshold, they are selected as new entries of type A and placed in the first column of the CIS directly and evaluation of $L(i)$ is not performed. If at least a one of offspring is larger than (or equal to) the current operational threshold, evaluation of $L(i)$ is performed. In this case, if $L(i)$ is insignificant, a zero is sent, the type B set is removed from the second column of the CIS, its position number and its type are preserved until the next level of compression. If $L(i)$ is significant, a one is sent, its two offspring are selected as new entries of type A and placed in the first column of the CIS. Then, the type B set is removed from the second column of the CIS. When all sets in the second column of the CIS were checked, the coding algorithm starts to test new type A sets in the first column of the CIS which were previously created (during evaluation of each type B set in the previous part, these new type A sets were created). The sorting pass of the CIS is finished when there is not any new type A set in the first column of the CIS, i.e. it was created no new type A set from evaluation of type B sets in the second column of the CIS.
After the sorting pass of the CIS, ***refinement pass*** is started. The coding algorithm starts to check each old entry of the CSP (the location of the coefficients which were significant under the last operational threshold). For the first level of compression, there is not any old entry in the CSP. In the refinement pass, if each old entry of the CSP is significant under the current operational threshold, a one is sent, and if it is insignificant under the current operational threshold, a zero is sent. Then, the magnitude of each coefficient which its position number is in the CSP, is reduced by the current operational threshold. Also, if the magnitude of a coefficient which its position number is in the CSP, is smaller than the current operational threshold, that magnitude is not reduced. After the refinement pass, the current operational threshold is halved, the sorting passes and the refinement pass are repeated until the desirable bit rate or quality requirement is reached.
For the next level of compression, reserved information from the previous level of compression is considered. This information includes: the position number of insignificant coefficients in the previous CIP, the location and type of insignificant sets in the previous CIS, and the position number of significant coefficients in the previous CSP. For the next level of compression, the halved operational threshold is compared to the threshold of each subband, and this comparison is performed like the previous coding level but with a small difference. Comparison is stopped as soon as comparison reaches the wavelet subband which the position number of its last coefficient has been selected as the frontier in the previous coding level. Also, if the new operational threshold does not equal any threshold of subbands, the frontier will not change.

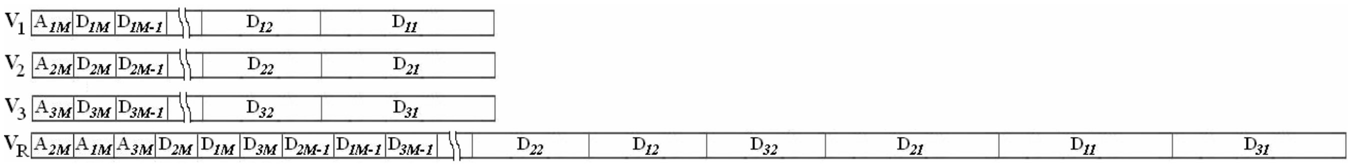

Fig.3. The formation way of the vector R (V_R); V_1, V_2, V_3 are vectors of the wavelet coefficients for channel 1, channel 2 , and channel 3, respectively

Regarding Example I, the frontier of next coding levels are shown in Table I.

TABLE I
THE FRONTIER CORRESPONDING TO THE OPERATIONAL THRESHOLD OF EACH CODING LEVEL IN EXAMPLE I

Operational Threshold	512	256	128	64	32	16	8	4	2	1
Frontier	7	7	15	15	31	31	31	31	63	63

As to be explained, the frontier limits redundant evaluation in the sorting pass of the CIS, i.e. evaluation which conduces to find the significance of *D*(*i*) (or *L*(*i*)), is performed as far as any position number of coefficients which belong to *D*(*i*) (or *L*(*i*)), is not larger than the frontier, otherwise evaluation of the significance of *D*(*i*) (or *L*(*i*)) is stopped. This fact helps us to decrease computation. Whereas checking tree structures takes most of time in the encoder part, this action decreases the time of compression significantly.

Furthermore, it is mentioned that before evaluation of the significance of *L*(*i*), the two offspring which are direct children of a tree node at location (*i*), are considered. If these two offspring are smaller than a given operational threshold, in the tree structure corresponding to location (*i*), a significant coefficient (or significant coefficients) is (or are) in the lower subtree, and coefficients in locations (*2i*) and (*2i+1*) will be roots of new type *A* sets definitely. Therefore, it is not necessary to evaluate *L*(*i*) and send a bit for *L*(*i*). Hence, evaluation of *L*(*i*) which is time-consuming, is not performed and computation is decreased. Due to sending a bit for evaluation of each *L*(*i*), if this evaluation is not performed, no bit is sent. Overall, at the same PRD, the ESPIHT algorithm can have performance faster than the SPIHT algorithm and the length of a transmitted bit stream is generally reduced.

IV. MULTICHANNEL ECG COMPRESSION

In this section, the multichannel compression method based on the ESPIHT algorithm is explained. In this method, first, the wavelet transform is implemented for each channel. Then, a new vector which is called V_R , is constructed with the wavelet coefficients of ECG channels. Afterwards the ESPIHT algorithm begins encoding the new vector V_R . The new vector helps us to have better compression because more appropriate tree structures are formed, and the new vector causes more efficient compression.

As in [9], wavelet coefficients with larger magnitude are generally placed in low frequency subbands. In order to compress multichannel ECG signals, it is better, the subbands of ECG channels are appropriately arranged. For compression of multichannel ECG signals, first, the subbands of channels are arranged, in order to arrange the subbands, the largest threshold from each channel is determined, this threshold is calculated such as the initial operational threshold in the ESPIHT algorithm. Then, the length of each subband for each channel is determined. Among ECG channels, a channel with the largest threshold is selected (initial channel). Then, another channel which has a threshold larger than residual channels but its threshold is smaller than the previous selected channel (initial channel), is selected. Other channels will be selected by this way. The wavelet coefficients of a selected channel will be placed in the new vector *R*. I use a simple example to show how this procedure works (Fig.3). The number of ECG channels is three, a *M*-level wavelet decomposition is performed, and the length of ECG signals is the same for all channels. The largest threshold of the wavelet coefficients for channel 1 is 256, for channel 2 is 512, and for channel 3 is 128. The length of each subband for a channel is determined according to equation (1). The channel 2 is selected, because it has the largest threshold. After that, its approximation coefficients (A_{2M}) are placed in the vector *R*. Then, the approximation coefficients of channel 1 are placed in the vector *R*, afterwards the approximation coefficients of channel 3 are placed. After placing the approximation subbands of all channels in the vector *R*, the detail subbands of channels will be placed in the vector *R*, thus the *M*th detail subband of channel 2 (D_{2M}) is placed in the vector *R*, then, this action is performed for channel 1 and channel 3. This procedure continues until the first detail subband of the channel 3. The location of each coefficient in the vector *R* is determined according to the length of selected subbands. When all coefficients of channels were placed in the vector *R*, the ESPIHT algorithm begins encoding the vector *R* and encoding data is stored or transmitted.

In the decoder side, the same procedure is performed. After decoding a bit stream, the reconstructed vector *R* will be separated into three vectors, and each of them is the reconstructed vector of the wavelet coefficients for each channel. Reconstruction of the vector of a channel is performed according to placement of its coefficients in the vector *R*.

V. RESULTS

For showing efficiency of the ESPIHT algorithm, i used data from the MIT-BIH arrhythmia database. The sampling rate and the sample resolution are 360 samples/s and 11 bits/sample, respectively, and the sample values are in the range from -1024 to 1023. A five level wavelet decomposition is used and the biorthogonal 9/7 tap filters are selected for the wavelet filter bank. In order to evaluate the redundancy removing capability of coding algorithms, a typical criterion is the compression ratio (CR). The percent root mean square difference (PRD) and the compressed data rate (CDR) are used as the evaluation criteria for the fidelity of reconstructed ECG data and efficiency of an ECG compression method, respectively. The evaluation criteria can be formulated as

$$CR = \frac{B_{Original}}{B_{Total}} \tag{3}$$

and

$$PRD = \sqrt{\frac{\sum_{i=1}^{L}(x_i - \hat{x}_i)^2}{\sum_{i=1}^{L} x_i^2}} \times 100 \tag{4}$$

and

$$CDR = \frac{R_s \times B_{Total}}{L} \tag{5}$$

where x_i is the ith original data sample, $\hat{x}_i$ is its reconstructed data sample, $B_{Original}$ is the number of original bits, B_{Total} is the total number of compression bits to be transmitted or stored, R_s is the sampling rate, and L is the data size. Average percent root mean square difference (APRD) is used to evaluate the reconstructed signals in multichannel compression [10].

A. The Simulation Results of the ESPIHT Algorithm

The proposed algorithm was tested using 14 selected records from the MIT-BIH arrhythmia database. This dataset includes 1-min length of data in record numbers 100, 107, 111, 112, 115, 116, 117, 118, 119, 121, 207, 213, 214, and 231. The average PRD results at different CR's are presented in Table II.

TABLE II
AVERAGE TEST RESULTS FOR THE FIRST DATASET

CR	4:1	6:1	8:1	10:1	12:1	16:1	20:1	24:1
PRD	1.01	1.38	1.81	2.18	2.57	3.52	4.82	6.47

In comparison with [3], the results of the ESPIHT algorithm are better than those of the SPIHT algorithm. They reported the PRD value of 5% at bitrate 183 bps for record 119. The PRD value of the coder proposed here is 3.97% for the same record and bitrate. The summery of this comparison is presented in Table III.

TABLE III
PRD COMPARISON BETWEEN THE ESPIHT ALGORITHM AND THE SPIHT ALGORITHM FOR RECORD 119

Algorithm	PRD(%)	CDR(bps)	Sampling Rate(Hz)	Bits/sample
ESPIHT	3.97	183	360	11
SPIHT	5	183	360	11

In order to demonstrate the effect of the frontier in the ESPIHT algorithm, 5-min (108000 samples) of the original signal from record 117 was selected. In this case, the whole number of coefficients which belong to type *A* and type *B* sets in the ESPIHT algorithm and the SPIHT algorithm too, are presented in Table IV. In this experience, in order to initialize the CSP, the CIP, and the CIS, i used the rule which has been explained in section III. The length of the approximation subband is also 3375.

TABLE IV
THE EFFECT OF FRONTIER ON THE NUMBER OF DESCENDANTS

Operational Threshold	Frontier	Number of Descendants in ESPIHT	Number of Descendants in SPIHT	PRD's
1024	3374	1	104626	71.96
512	3374	1	104626	24.8
256	13499	10124	104624	12.9
128	26999	22742	103742	7.93
64	26999	22018	103018	4.36
32	53999	47908	101908	2.8
16	107999	100898	100898	2.05
8	107999	98708	98708	1.34
4	107999	85750	85750	0.85

In Table IV, the frontier value, the whole number of descendants which belong to type *A* and type *B* sets in the ESPIHT algorithm, also, the whole number of descendants which belong to type *A* and type *B* sets in the SPIHT algorithm at the beginning of each level of compression, finally, the PRD value at the end of each level of compression, are presented corresponding to the operational threshold of each level of compression. According to Table IV, the number of descendants which are checked to find the significance of type *A* and type *B* sets in the ESPIHT algorithm, is less than the SPIHT up to the seventh level of compression, thus the time which is spent for checking these descendants, is significantly decreased.

Overall, the speed of computation is increased because of the enhanced sorting pass of insignificant sets.

There are other wavelet coding algorithms in the literature. Hilton presented a compression method based on the EZW algorithm [4]. In order to brief comparison with this coder, and the SPIHT algorithm, the record 117 was used again. As reported in [3], the PRD value of 1.18% with CR 8:1 for record 117, was presented and compared with the PRD value of 2.6% for the same record and CR reported in [4]. The PRD value of the coder proposed here is 1.02% for the same record and CR.

B. The Results of the Multichannel Compression Method

In this part, the results of multichannel ECG compression by the multichannel compression method based on the ESPIHT algorithm are presented. The simulation results for dual channel ECG are listed in Table V. The results are based on 14 records from the MIT-BIH arrhythmia database (1-min length of data in record numbers 100, 104, 107, 111, 112, 115, 116, 117, 118, 119, 121, 213, 214, and 231).

In comparison with the single channel SPIHT or ESPIHT, the multichannel compression method based on the ESPIHT algorithm has performance faster than them, because arranged wavelet subbands of input channels cause to form more appropriate parent-offspring relations between wavelet coefficients of lower frequency subbands at the temporal orientation trees. According to experiments, the results of the multichannel compression method at high CDR's are almost the same with the single channel ESPIHT but this method will have appropriate results at high CR's or low CDR's.

TABLE V
THE RESULTS OF THE MULTICHANNEL COMPRESSION METHOD BASED ON THE ESPIHT ALGORITHM FOR THE SECOND DATASET

Record	APRD at Bitrate 396 (bps)	APRD at Bitrate 282.8 (bps)	APRD at Bitrate 247.5 (bps)	APRD at Bitrate 220 (bps)	APRD at Bitrate 198 (bps)	APRD at Bitrate 165 (bps)
100	3.11	3.72	4.27	4.64	5.45	6.64
104	3.96	5.48	6.38	7.37	8.47	10.22
107	2.05	3.13	3.56	4.47	5.47	6.99
111	4.07	4.83	5.19	5.59	6.07	7.67
112	1.42	1.96	2.19	2.34	2.64	3.14
115	2.97	3.86	4.29	4.54	5.03	6.16
116	1.72	2.44	2.76	3.01	3.51	4.45
117	1.71	2.32	2.51	2.61	2.93	3.44
118	2.28	3.32	3.73	4.4	4.99	6.21
119	1.6	2.23	2.49	2.86	3.16	4.3
121	1.32	1.73	1.89	2.04	2.18	2.76
213	2.45	3.99	5.15	5.96	7.19	10.6
214	2.66	3.33	3.53	4.08	4.56	6.07
231	3.52	4.68	5.35	5.78	6.86	8.54

In order to increase efficiency of compression regarding one signal which is stored for a long time, the multichannel compression method can be utilized. For demonstration of this case, i used 2-min of the original signal from the lead II of the record 100. This signal is recorded and divided into two parts, the first part with 1-min length is supposed as channel 1 and the second part with 1-min length is supposed as channel 2. The results of performance comparison between the multichannel compression and the single channel compression based on the ESPIHT algorithm, are listed in Table VI.

TABLE VI
PERFORMANCE COMPARISON BETWEEN MULTICHANNEL COMPRESSION AND SINGLE CHANNEL COMPRESSION FOR THE LEAD II OF THE RECORD 100

CR	4:1	8:1	12:1	16:1	20:1	24:1	28:1
Multichannel APRD's	1.3	2.32	2.81	3.7	4.96	6.21	8.13
Singlechannel PRD's	1.26	2.33	2.82	3.75	5.31	6.89	9.39

According to experiments, compression of one signal by the multichannel compression method will have appropriate results at high CR's or low CDR's.

VI. CONCLUSION

In this paper, i proposed the ECG data compression algorithm which is named ESPIHT. It is a modified version of the SPIHT algorithm. At the same PRD, the proposed algorithm has performance better than the SPIHT algorithm, because the evaluation way of the wavelet coefficients in the sorting pass of insignificant sets has been enhanced, moreover, if the execution time of compression is first priority in work (especially in telemedicine or home monitoring), the 1-D ESPIHT algorithm will be an appropriate choice in comparison with 2-D ECG compression methods which can be found in the literature [11]. Also, the proposed algorithm reduces the number of bits in a bit stream which is stored or transmitted. Furthermore, i presented a specific procedure based on the ESPIHT algorithm for multichannel ECG compression. I applied the proposed algorithm on selected records from the MIT-BIH arrhythmia database. The experimental results confirmed that the ESPIHT coding algorithm performs more efficiently than the SPIHT algorithm. Also, the multichannel compression method based on the ESPIHT algorithm can be utilized to improve compression results of a signal which is stored for a long time.

ACKNOWLEDGMENT

The auther would like to thank Prof. Mohammad Pooyan, Shahed university, Tehran, Iran, who encouraged me to work on wavelet compression of ECG signals.

REFERENCES

[1] L. J. Thomas, K. W. Clark, C. N. Mead, K. L. Ripley, B. F. Spenner, and G. C. Oliver, "Automated cardiac arrhythmia analysis," *Proc. IEEE*, vol. 67, pp. 1322-1337, Sept.1979.

[2] N. V. Thakor, "Ambulatory arrhythmia monitoring: From Holter monitors to automatic implantable defibrillators," *IEEE Trans. Biomed. Eng.*, vol. BME-31, pp. 770-778, Dec.1984.

[3] Z. Lu, D. Y. Kim, and W. A. Pearlman, "Wavelet compression of ECG signals by the set partitioning in hierarchical trees algorithm," *IEEE Trans. Biomed. Eng.*, vol . 47, pp. 849-856, July 2000.

[4] M. L. Hilton, "Wavelet and wavelet packet compression of electrocardiograms," *IEEE Trans. Biomed. Eng.*, vol. 44, pp. 394-402, May 1997.

[5] Y. Zigel, A. Cohen, and A. Katz, "ECG signal compression using analysis by synthesis coding," *IEEE Trans. Biomed. Eng.*, vol. 47, pp. 1308-1315, Oct. 2000.

[6] S. G. Miaou, and S. N. Chao, "Wavelet-based lossy to lossless ECG compression in a unified vector quantization framework," *IEEE Trans. Biomed. Eng.*, vol. 52, pp. 539-543, Mar. 2005

[7] N. V. Thakor, J. G. Webster, and W. J. Tompkins, "Estimation of QRS complex power spectra for design of a QRS filter," *IEEE Trans. Biomed. Eng.*, vol. BME-31, pp. 702-706, Nov. 1984.

[8] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 205-220, Apr. 1992.

[9] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern analysis and machine intelligence*, vol. 11, pp. 674-693, July 1989.

[10] S. Jalaleddine, C. Hutchens, R. Strattan, and W. Coberly, "ECG data compression techniques-A unified approach," *IEEE Trans. Biomed. Eng.*, vol. BME-37, pp.329-343, Apr.1990.

[11] S. C. Tai, C. C. Sun, and W. C. Yan, "A 2-D ECG compression method based on wavelet transform and modified SPIHT," *IEEE Trans. Biomed. Eng.*, vol. 52, pp. 999-1008, June 2005.