COMPRESSION AND DECOMPRESSION OF TIME SERIES SIGNALS USING TWO WAVELET COEFFICIENTS THRESHOLDING TECHNIQUES AND THEIR COMPARISONS

Oinam Suchitra Devi¹, Hemanth Kumar P.², S Basavaraj Patil³

¹M. Tech. IV Semester, DEC, AMC Engineering College, Bangalore, India.suchitra.oinam@gmail.com
²Asst. Prof., Dept. Electronics & Communication, AMCEC, Bangalore, India.hemanth00kumar@gmail.com
³Prof. & Head, Computer Science, BTLITM, Bangalore, India.dr.sharan9@gmail.com

ABSTRACT

A time series is a sequence of data point which is measured through repeated measurements over uniform time intervals. Compression can be defined as the reduction in size of data in order to save space or transmission time. For data transmission, compression of time series signal, like ECG or EEG, can be performed on just the data content or on the entire transmission unit depending on a number of factors. The design of time series compression schemes involves trade-offs among various factors which includes the degree of compression, the amount of distortion introduced and the computational resources required to compress and decompress the time series data. Data decompression is the process of reversing compressed data into its original or near to original so that it can read or seen as original. In general there exists measures that show how much reconstructed signal is similar to the raw one and the most used are Percentage Root mean square Difference (PRD), Peak Signal to Noise Ratio (PSNR) etc. This paper analyzes two compression methods Global thresholding of coefficients and fixed encoding compression method and Global Thresholding of coefficients and Huffman encoding compression method. From the results it is observed that Global Thresholding of coefficients and Huffman encoding compression method gives better performance as it achieve high compression ratio for smaller PRD and high PSNR with better quality of compression.

KeyWords
Fixed encoding, Global Thresholding, Huffman encoding, Time Series Signal.
INTRODUCTION

The need for an efficient technique for compression of time series signal in today’s world ever increasing because the raw or the original signals need large amounts of space and it seems to be a big disadvantage during transmission and storage. Compression is very useful as it helps reduce resources usage, such as data storage space or transmission capacity. Because compressed time series data must be decompressed to use, this extra processing imposes computational or other costs through decompression. For instance, a compression process for video may require expensive hardware for the video to be decompressed fast enough to be viewed because it is being decompressed. The design of time series data compression process involves trade-offs among various factors, including the degree of compression, the amount of distortion introduced and the computational resources required to compress and decompress the time series data.

Generally, compression methods can be categorized into lossless and lossy compression techniques. Lossless compression algorithms perform compression without any loss of information from the original one, e.g. for medical records like ECG and EEG, whereas the lossy compression algorithms provide larger compression ratios but when the previous original information is reconstructed, some information is lost. Most common and widely used lossless time series compression algorithms are Huffman encoding, LZW and arithmetic coding methods. Lossy compression methods are widely used in video conferencing and broadcast television, etc where certain loss of information is tolerable for the high rates of compression.

Decompression is the process of expanding a compressed data back to its raw form. It is the reconversion of compressed data into its original or near to original form so that it can be heard, read or seen as original. The main advantages of the time series data compression are less disk space as more data in reality, faster reading and writing, faster file transfer etc., but it has certain disadvantage like it added complication, effect of error in transmission, slower for sophisticated methods, need to decompress all previous data. Often signals to be compress are in the time-domain, but in order to process them more easily other information such as frequency domain is required. A good example for this idea cites the problem of multiplying two roman numerals. In order to do the calculation we would find it easier to first translate the numerals to number system, and then translate the number system back to a roman numeral. The result is the same, but taking the detour into an alternative number system made the calculation easier and quicker.

DATA DESCRIPTION

Table 1 shows the description of the data set 1 extracted from “Dictionary-Based Compression for Long Time-Series Similarity”, IEEE paper 2010. Table 2 shows the description of own data set which is refered as data set 2. It is equivalent to an Electrocardiogram (ECG) signal. The ECG signal must be filtered or smoothed with an N-point smoother, Savitzky-Golay FIR filter. Here y, y1 and y2 represents the Savitzky-Golay Filtering. Savitzky-Golay Filtering sgolayfilt (k,f,x) smoothes the signal x using a Savitzky-Golay smoothing filter. The polynomial of order k must be less than the frame size, f, and f must be odd number. The length of the input x must be greater than or equal to f. If x is a matrix then filtering is done on the columns of x. f=3, 5, 9. If the polynomial order k equals f-1 then no smoothing will occur. Both the data set are time series signal i.e. collection of observations of well-defined data obtained through repeated measurements over a uniform time interval.

### TABLE.1 DATA SET 1 DESCRIPTION

<table>
<thead>
<tr>
<th>Data</th>
<th>Value</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>121×3181 double</td>
<td>0</td>
<td>29010...</td>
</tr>
<tr>
<td>y</td>
<td>1×121 double</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>

### TABLE.2 DATA SET 2 DESCRIPTIONS

<table>
<thead>
<tr>
<th>Data</th>
<th>Value</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>500x1 double</td>
<td>-0.8390</td>
<td>1</td>
</tr>
<tr>
<td>y</td>
<td>500x1 double</td>
<td>-0.7912</td>
<td>0.9583</td>
</tr>
<tr>
<td>y1</td>
<td>500x1 double</td>
<td>-0.5712</td>
<td>0.7734</td>
</tr>
<tr>
<td>y2</td>
<td>500x1 double</td>
<td>-0.7529</td>
<td>0.9249</td>
</tr>
</tbody>
</table>
METHODOLOGY

The compression and decompression of time series signal is implemented in these steps:

Step 1) Time series signal like ECG, EEG or stock market etc is generated and stored in the database.

Step 2) Apply the two compression methods.

i) Global Thresholding of coefficients and Fixed encoding

ii) Global Thresholding of coefficients and Huffman encoding

Steps for data compression:

A) Decompose

Choose a wavelet, choose a level n. Compute the wavelet decomposition process of the signal at level n.

B) Threshold detail co-efficient

For each level from 1 to n, a threshold level is selected and applied global thresholding to the detail.

C) Quantization

Fixed or Huffman coding can be used for the quantization process depending on the method.

D) Reconstruct

Compute wavelet reconstruction using the original approximation coefficient of level n and modified the detail coefficient of level from 1 to n.

Step 3) Decompress: In order to compressed data back to its original form, performed decompression.

Step 4) Compare the result in various comparing measures that show how much reconstructed signal is similar to the original one.

Wavelet Coefficients Thresholding

In Coefficients Thresholding Methods there are two parameters which may be used. The first one is related to the threshold and the second one is the number of classes for quantization. Thresholding is the most commonly used processing tool in wavelet multi-resolution analysis. The wavelet thresholding method was mainly developed for removing noise and outliers, compression, and pattern recognition of the data signal before wavelet reconstruction. Thresholding is a simple and non-linear method, which is operated on one wavelet coefficient at a time of period. In the basic form, each of the coefficients is thresholded by comparing against one threshold, if the coefficient is smaller than the threshold level, it set to zero, and otherwise it is kept as it is or modified. Replacing the small noisy coefficients by zero and the inverse wavelet transform on the result may lead to reconstruction with the essential data characteristics, with less noise and better quality of image.
The basic ideas presented in true compression methods is that which cascade in a single step, coefficient Thresholding i.e global or by level and encoding by quantization. Here Global thresholding of coefficients is used in both the compression method. Fixed or Huffman coding can be used for the quantization of the signal depending on the method of compression used. The notion behind the signal data compression is based on the concept that the regular signal component can be accurately approximated by using the elements and a small number of approximation coefficients and some of the detail coefficients. Wavelet analysis can be used to divide the information of signal data into approximation and detail subsignals. If the detail subsignals are very small then they can be set to zero without significantly changing the data. The value below which the details are considered small enough to be set to zero is called as the threshold. The greater the number of zeros the greater the signal data compression that can be achieved. The amount of information retained after compression and decompression by a data is called as the ‘energy retained’ and this is proportional to the sum of the squares of the pixel values. If the energy retained is 100% then the compression is known as lossless as the data can be reconstructed exactly as the raw one. This occurs when the threshold is set to zero value that means the detail has not been changed. If any values are changed then energy will be lost and this is known as ‘lossy’ compression. Ideally, during time series compression the number of zeros and the energy retention will be as high as possible. However, as more zeros are obtained more energy is lost during data compression.

**Global Thresholding of coefficients and Fixed encoding**

In time series data Compression, we addressed the aspects specifically related to compression using wavelets. However, in addition to the algorithms which is related to wavelets like DWT and IDWT, it is necessary to use other ingredients concerning the quantization mode and the coding type in order to deal with true data compression. The basic ideas presented above statement are used by three methods which is cascaded in a single step i.e coefficient thresholding, it may be global or by level, and encoding by quantization. Fixed or Huffman coding can be used for the quantization depending on the method of data compression. Global thresholding consists of setting an intensity value known as threshold such that all voxels having intensity value below the threshold belong to one phase and the remainder belong to the other phase.

A fixed-length encoding is an encoding such as ASCII and is convenient because the boundaries between characters are easily determined and the pattern used for each character is completely fixed for example ‘a’ is always exactly 97. The standard ASCII character encoding uses 8 bits amount of space to store each character. Common characters don’t get any special treatment and they require the same 8 bits that are used for much rarer characters such as ‘ü’. For example a file of 1000 characters encoded using the ASCII scheme will take 1000 bytes i.e 8000 bits and it has no more, no less, whether it be a file of 1000 spaces to a file containing 4 instances each of 250 different characters. In practice, it is not the case that all 256 characters in the ASCII set occur with same frequency. In an English text document, it might be the case that only 90 or so distinct characters are used at all that means 166 characters in the ASCII never even appear and within those 90 characters there are likely to be significant differences in the character counts. Fixed length encoding uses the same number of bits for each symbol and k-bit code supports 2^k different symbols.

**Global Thresholding of coefficients and Huffman encoding**

The Huffman encoding algorithm is an optimal data compression algorithm when only the frequency of individual letters is used to compress the data. The idea behind the said algorithm is that if we have some letters that are more frequent than others, it makes sense to use fewer bits to encode those letters than to encode the less frequent letters. Huffman coding is an entropy encoding algorithm and is used for lossless data compression. The term refers to the use of a variable length code table for encoding a source symbol such as a character in a file where the variable length code table has been derived in a particular way and is based on the estimated probability of occurrence for each possible value of the source symbol. The Huffman encoding process takes advantage of the disparity between frequencies and it uses less storage for the frequently occurring characters at the expense of having to use more storage for each of the more rare characters.

Huffman encoding is an example of a variable length encoding such that some characters may only require 2 or 3 bits and other characters may require 7, 10, or 12 bits. The savings from not having to use a full of 8 bits for the most common characters that makes up for having to use more than 8 bits for the rare characters. The overall effect is that the file almost always requires less data space. Huffman encoding uses different number of bits to encode different characters. The original symbols from a compressed file are replaced with bit strings and the more frequently a given symbol appears in the compressed file and the shorter bit string for representing the symbol. The encoded symbols and their corresponding bit strings are represented as a Huffman tree and the Huff-
man tree is used for both compressing and decompressing.

Quantitative and Perceptual Quality Measures

Two measures are use to measure the quantitative of the compression performance. One of the measure is the compression ratio CR, which means that the compressed data is stored using only CR% of the initial storage size. Data compression ratio, also called as compression power, is a computer-science term and is used to quantify the reduction in data-representation size produced by a data compression algorithm. The data compression ratio is analogous to the physical compression ratio and is used to measure physical compression of substances. Data compression ratio can be defined as the ratio between the compressed size and the uncompressed size of the data and is given by

$$\text{CR} = \frac{\text{Compressed size}}{\text{Uncompressed size}}$$

Another measure is the Bit-Per-Pixel ratio BPP and which gives the number of bits required to store one pixel of the image. For a gray scale image, the initial BPP is 8 while for a true colour image the initial BPP is 24 as 8 bits are used to encode each of the three colours i.e Red, Green and Blue colour. The challenge of the data compression techniques is to find the best compromise between a weak compression ratio and a good perceptual result.

There are several measures to evaluate the perceptual quality of the time series signal. MSE is the Mean Square Error and it represents the mean squared error between the compressed and the original data. It measures the average of the squares of the errors. The error can be defined as the amount by which the value implied by the estimator differs from the quantity to be estimated. The difference occurs due to randomness or as the estimator doesn’t account for information that could produce a more accurate estimate. MSE is given by:

$$\text{MSE} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} (X(i,j) - Xc(i,j))^2$$

Where X(i,j) and Xc(i,j) are the original data and its corresponding compressed data. The lower the value of MSE, the lower the error of the compression.

PSNR is the Peak Signal to Noise Ratio and it represents a measure of the peak error. It is usually expressed in decibels. PSNR is most commonly used signal to noise ratio to measure the quality of reconstruction of lossy compression. The signal in this case is the original time series and the noise is the error introduced by data compression. The higher PSNR generally indicates that the reconstruction is of higher quality. PSNR is defined by:

$$\text{PSNR} = 10 \log_{10} \left( \frac{255^2}{\text{MSE}} \right)$$

The higher the value of PSNR, better the quality of the compressed or reconstructed time series data. Typical values for lossy compression of a data are between 30 and 50 dB and when the PSNR is greater than 40 dB, then the two data are indistinguishable.

Also there exist several other measures to know the quality of the signal and one of the popularly used measure is Percentage Root mean square Difference (PRD). PRD provides a numerical measure of the root mean square (rms) error. This parameter as quality measurement can mask the real performance of an algorithm as the PRD depends a lot on the mean value of the raw signal. It is given by

$$\text{PRD} = 100 \times \sqrt{\frac{\sum_{i=1}^{l} \sum_{j=1}^{n} \left( X(i,j) - Xc(i,j) \right)^2}{\sum_{i=1}^{l} \sum_{j=1}^{n} X(i,j)^2}}$$
Where \( x(i) \) and \( \hat{x}(i) \) are the \( i^{th} \) sample of the original and reconstructed time series signal of length \( N \).

Maximal absolute difference (MAD) is the maximum absolute difference that measure between the original and the compressed or decompressed data. As PRD does not show exactly how much signal is distorted in different time position, but shows only the cumulative distortion. Very often Maximal Absolute Difference is used to measure the difference between reconstructed and raw signal.

**Results and Comparison**

The result of time series compression is shown below. Fig 2 and Fig 3 shows the Time Series Compression by Global Thresholding of coefficients and Fixed encoding techniques in data set 1 and data set 2 respectively and it shows the difference in the original and reconstructed signal. Fig 3 and Fig 4 shows the time series compression by Global Thresholding of coefficient and Huffman encoding in data set 1 and 2 respectively and also it shows the difference in original and reconstructed signal. Table 3 and Table 4 shows detail of the quantitative and perceptual quality measures of the compressed data set 1 and data set 2 respectively. From the table we observed that Global Thresholding of coefficients and Huffman encoding has its CR value high for smaller PRD, high PSNR with better quality of compression and its MAD value is less, compared to the Global Thresholding of coefficients and Fixed encoding. So, Global Thresholding of coefficients and Huffman encoding compression method is better in compression compared to Global Thresholding of coefficients and Fixed encoding compression method.

A) Time series compression by Global Thresholding of coefficients and Fixed encoding

![Fig 2: Data set 1 compression by Global Thresholding of coefficients and Fixed encoding](image)

B) Time series compression by Global Thresholding of coefficients and Huffman encoding

![Fig 3: Data set 2 compression by Global Thresholding of coefficients and Fixed encoding](image)

![Fig 4: Data set 1 compression by Global Thresholding of coefficients and Huffman encoding](image)
TABLE 3: CR, BPP, MSE, PRD and PSNR FOR THE DATA SET1

<table>
<thead>
<tr>
<th>Methods</th>
<th>CR</th>
<th>BPP</th>
<th>MSE</th>
<th>PRD</th>
<th>PSNR</th>
<th>MAD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global thresholding of Coefficients and Fixed encoding</td>
<td>6.7677</td>
<td>0.3414</td>
<td>1.1745e+04</td>
<td>0.0150</td>
<td>87.3730</td>
<td>7.4220</td>
</tr>
<tr>
<td>Global thresholding of Coefficients and Huffman encoding</td>
<td>28.0000</td>
<td>2.6000</td>
<td>1.1025e-04</td>
<td>0.0105</td>
<td>87.7068</td>
<td>0.0961</td>
</tr>
<tr>
<td>Global thresholding of Coefficients and Huffman encoding</td>
<td>7.3673</td>
<td>0.5894</td>
<td>7.1236e+03</td>
<td>0.0079</td>
<td>90.1545</td>
<td>0.0667</td>
</tr>
</tbody>
</table>

TABLE 4: CR, BPP, MSE, PRD, PSNR and MAD FOR THE DATA SET 2

Conclusion

In this paper, two Wavelet Coefficients Thresholding compression Techniques is implemented. Two compression methods Global thresholding of coefficients and fixed encoding compression method and Global thresholding of coefficients and Huffman encoding compression method are discussed. The results are presented on different data set of time series signals of varying characteristics. From the results it is observed that Global thresholding of coefficients and Huffman encoding compression method gives better performance in compression. Compression of time series signal can be fundamental not only for the obvious storage size reduction but also improving its performance. This is because of the less data needs to be read or written on disk stored in a block device. The work presented in the paper may be helpful for the design of efficient ECG compressor and EEG compression that can be useful for many hospitals.

References


