A Similarity Search Method of Time Series Data with Combination of Fourier and Wavelet Transforms

Kyoji Kawagoe and Tomohiro Ueda

Computer Science Department
Ritsumeikan University
1-1-1, Noji-Higashi,
Kusatsu-city, Shiga
JAPAN
kawagoe@cs.ritsumei.ac.jp, ueda@ims.cs.ritsumei.ac.jp

Abstract

Recently, time-series data, such as stock exchange rates and weather data, has widely been used in many fields. Similarity search of time-series data is important because it is useful for predicting data changes and for searching for common sources. In this paper, we propose a new similarity search method of time-series data using both Discrete Fourier Transform (DFT) and Wavelet Transform (WT). A method of reducing time-series indexing size, using a correlation coefficient, is also presented.

1. Introduction

At present, time series data that changes with a change of a time, such as stock price, exchange rate, weather data, accounting data, and POS data, are used in many fields. Time series data is useful for the estimation of the future data, by comparing time series data from a viewpoint of a similar characteristic. For example, it is possible that changes in the stock price of a certain company can be estimated from changes of other companies which are similar to the change of the company in the past.

In order effectively to make use of time series data, it is important to employ effective and efficient similar search methods with which some time series data much similar to the given time series data can be obtained . In order to do a similar search, at first, they are necessary to define whether two time series data are alike or not. Usually, Euclid distance is used as a basis of measuring similarity. However, the calculation of the distance for a large amount of time series data is time consuming. In order to reduce the computation, there have been many similarity search and indexing methods for time-series data.

The well-known method among the existing similarity search methods is the use of Discrete Fourier Transform (DFT) [2,6,7]. The co-efficient of the DFT is calculated and can be used for time-series data indexing. Recently, a similar search method by means of Wavelet transform (WT) [1,4,5] is also proposed.

The DFT and the WT have different characteristics, which the filtering result of each is quite different. The DFT can select the similar time series data which can not be selected by the WT, while the DFT fails to select the similar time series data which the WT can select. In order to obtain more effective search result, in this paper we propose a new searching method combining both the DFT and WT.

In searching time series data, it is necessary to obtain a partial matching result, meaning that the given time-series data is similar to a part of some time series data stored in the databases. The processing is more time-consuming and requires much amount of indexing data. Reducing indexing size is an important problem to be solved. In this paper, we also present an indexing reduction method for efficient partial matching of time series data.

The rest of the paper is organized as follows. Section 2 gives the overview of the similarity search of the time series data. Section 3 presents a new similarity searching method with combination of the DFT and WT. In the Section 4, some techniques of the partial matching and reduction of the indexing size is presented. The experimental results for comparing our proposed method, the DFT method and the WT method are given in Section 5. Section 6 explains some applications of the proposed method. The conclusions are given in Section 7.

Proceedings of the Ninth International Symposium on Temporal Representation and Reasoning (TIME'02)
1530-1311/02 $17.00 © 2002 **IEEE**

2. Similarity Searching of Time Series Data

2.1. Similarity

It is important to define the similarity measure showing similarity between two time series data. Here, the similarity measure means that the larger the measure is, the more the shapes of the two time series data are not resembled. In this paper, we use Euclid distance as a basis of calculating similarity measure, as usual.

Suppose that there are two time series data such as $A=\{a_1,a_2,\ldots a_n\}$ and $B=\{b_1,b_2,\ldots b_n\}$. The Euclid distance E and the similarity measure S are respectively defined as follow:

$$E=\sqrt{\sum_{i=1}^{n}(a_i-b_i)^2} \qquad S=1/(E+1) \quad 0<S\le 1$$

With the above Euclid distance, there are some problems in applying the distance for time-series data search. One of them is its computational complexity. The more the number of time-series data N is increasing, the more the distance calculation for searching is increasing as $O(N^2)$. Another problem lies in difficulty of indexing. Little efficient direct indexing technique for similarity search of time-series data has been presented.

2.2. Existing Similarity Search of Time Series Data

Besides direct indexing of time-series data, a lot of indexing techniques has been presented. Most of the work uses a transformation from the time dimension into some feature space. For example, in order to check the similarity between two time series data called $A=\{a_1,a_2,\ldots a_n\}$ and $B=\{b_1,b_2,\ldots b_n\}$, the two sequences of data are transformed into the feature space When it is assumed that the feature transformation function is C(X) where X is a time series data, C(A) and C(B) are compared instead of comparing A and B. With guaranteeing no false dismissal, that is $D_F(C(A),C(B))\le D(A,B)$, a practical index size can be obtained, where D(.,.) and $D_F(.,.)$ are the distance function in the original space and the distance function in transformed space, respectively. .

The most representative method is use of Discrete Fourier Transform. This transform guarantees no false dismissals [7]. The Wavelet Transform is shown to guarantee no false dismissals when the Haar basis is used [1].

Besides these well-known transforms, other methods with dimensionality reduction include SVD decomposition [2], random projections [17], and FastMap [16].

3. Our Proposed Method

3.1. Basic idea

With combination of the two methods of DFT and WT, the computation of the search process in the Euclidean space can be reduced. That is, both of the DFT and WT are used as a kind of filtering, before checking the distance in the Euclidean space. The following process is used in our proposed method.

Before searching, indices constructed using the DFT and WT and applying these transforms for all the time series data set in the database. When a time series data is given for querying, the data is also transformed using these transforms.

At first, given a time series data as a query, a set of characteristics is calculated from the data using the DFT and the WT. Then, an index by using DFT and WT. Using the DFT and WT indexes of the database, some of the similar time series data are obtained. The selection is done by comparing the values of the given time series data characteristics and those of characteristics for each in the database. After selecting those, each obtained time-series data which is a candidate as a result and is satisfied with the query condition, is compared with the given time-series data in the Euclidean space. Finally, the closest time-series data in the Euclidean space is obtained. In the same way the second closest, the third closest, and so one are obtained in turn.

In the DFT and WT filtering phase of the above, the DFT is used first, the WT is used after the DFT filtering. It is because from observation of our many experiments the DFT filtering performance is superior to that of the WT. Especially; the WT filtering sometimes fails to select the much similar time series data to be selected.

The more detail is described in the 3.4.

3.2. Fourier Transform

DFT is a transform using a sin function and cos function from time space into frequency space. The DFT is described in the following.

When x(n) where n = 0,1...,N -1 as a time series data are given, the DFT k-coefficient X(k) is expressed as follow.

$$X(k)=\sum_{n=0}^{N-1}x(n)e^{-j2\pi kn/N}$$

By expanding this right side, the following is obtained.

$$X(k)=\sum_{n=0}^{N-1}x(n)\sin(2\pi kn/N)-j\sum_{n=0}^{N-1}x(n)\cos(2\pi kn/N)$$

The DFT characteristics are defined as the following *P(k)* and *Q(k)*.

$$P(k) = \sum_{n=0}^{N-1} x(n)\sin(2\pi kn / N)$$
$$Q(k) = \sum_{n=0}^{N-1} x(n)\cos(2\pi kn / N)$$

In the method presented in the paper, we introduce *P(0)*, *P(k)* and *Q(k)*, *k*=1, L as the DFT coefficients, where L is a parameter to be decided beforehand.

3.3. Wavelet Transform

In the Wavelet transform with Haar base, there are two kinds of functions called approximation function and difference function. The approximation function generates a sequence of the averages between two consecutive data in the input sequence. The difference function generated a sequence of the differences between two consecutive data in the current approximation sequence. These functions are applied recursively until the number of the elements in the difference sequence is one.

That is, the i-th approximation sequence A_i is defined as follow,

$$A_i = \{\frac{A_{i-1}(1)+A_{i-1}(2)}{2}, \frac{A_{i-1}(3)+A_{i-1}(4)}{2}, \ldots, \frac{A_{i-1}(m-1)+A_{i-1}(m)}{2}\}$$

where $A_i(j)$ is the j-th element in the sequence A_i and m is the number of the elements in the sequence A_{i-1} . The i-th difference sequence D_i is also defined as follow.

In the method presented in the paper, we introduce the coefficients of $D_K, D_{K-1}, \ldots D_{K-M+1}$ elements, where M is a parameter to be decided beforehand. As the number of elements in the (K-j+1)-th difference sequence is 2^{j-1}, the total number of the WT coefficients for indexing is 2^M+1.

3.4. Combination of the Fourier and Wavelet Transform

The DFT and WT filtering process is described before.

Step1: With the DFT, the time-series data which is not similar to the given time-series data. The similarity is checked whether the DFT coefficients of the checked data are within the small distance from the DFT coefficients of the given data or not. The only DFT similar data can be selected.

Step2: For all the data whose DFT coefficients are close to the given data DFT coefficients, its WT coefficients are compared with the given data WT coefficients. As in the same as the DFT filtering, the similarity is checked whether the WT coefficients of the checked data are within the small distance from the WT coefficients of the given data or not. The only WT similar data can be selected.

The reason why the DFT is processed forwarded by the WT is that from our experiment described in the Section 5 it is obtained the DFT is superior to the WT. More useful time-series data is discarded with the WT than with the DFT.

Step3: Using the Euclidean distance, the distance between the given data and each data selected in the step 2 is calculated. With the distance value, the time-series data is sorted and put as a query result.

4. Sub-sequence Matching

4.1. Sub-sequence Matching

Sub-sequence matching is to obtain some time series data sequence of which sub-sequence is similar to a part of time series data sequence given as a query. There are two basic methods to realize this matching. One is to decompose all the time series data sequences into s set of meaningful sub-sequences in advance. Then, the usual time series data retrieval is applied to a set of sub-sequences. The other is to shift a time series data on the time dimension one by one and to generate many sub-sequences from the data. This shift operation is applied to all the time-series data sequences of a database in advance. Then after indexing of these sub-sequences, the sub-sequence matching can be processed using the indices and one of the usual search methods.

From the search performance, the latter method is appropriate, while the latter contains several problems. The most important problem to be solved is to reduce the index size. In the shift method, the number of index entry is enormously increasing. In the next, we propose the use of the correlation coefficient.

4.2. Index Size Reduction

In our method, there are two kinds of the index entries: that is DFT coefficients and WT coefficients. Among the index size reduction method, we assume in general that there are index entries whose structure is in the form of

$$x = \{X(1), X(2), \ldots, X(n)\}.$$

The correlation coefficient R(x,y) is defined as follow.

$$R(x.y) = Cov(x, y) / S_x S_y$$

where $x = \{X(1), X(2), \ldots, X(n)\}$ and $y = \{Y(1), Y(2), \ldots, Y(n)\}$ are time series index data, Cov is a covariance between x and y, and S_x, and S_y are the standard deviation of x and y, respectively.

In order to construct the reduced-size indices, first, the correlation covariance for any pairs of time-series index data x and y is calculated. Then if the value of the correlation covariance is within some threshold, called

Clustering Threshold, set in advance, these two time series index data is supposed to be similar. Finally, combining a set of the similar time-series index data into one index entry, the size of the time-series index data can be reduced, as only one index data is selected as a representative for each the similar index data set.

5. Experimental Results

5.1. Experiment Environment

In order to show the efficiency of the method presented in the paper, we made several experiments as follow. In all the experiments, a set of stock data of companies whose stocks are listed on the Tokyo Stock Exchange is used. The interval of a sequence used is from July, 1996 to July, 2001. The number of the stock time sequence is 3000. A time series data for a query is selected in random for each experiment. During the experiments, we set L and M to 2 and 3, respectively.

Moreover, the allowable error range of similarity in the DFT and WT filtering is assumed to be within 2%.

The following ratios are used for evaluation.

1) Recall ratio: Recall ratio is
(The number of correct answers within the result) / (the number of correct answers in the database)
2) Precision Ratio Precision ratio is
(The number of correct answers within the result) / (the number of the result)
3) Index reduction ratio: Index reduction ratio is
(The number of index entries after clustering) /
(The number of index entries before clustering)

The values in the figures below are the average of the 40 times experiment result values.

5.2. Experiment I

In the Experiment I, the three methods, DFT, WT and DFT+WT are compared from the precision ratio viewpoint, with change in the number of the database size. The result is shown in the figure 1.

As shown in the figure 1, the DFT+WT, our proposed method, has the highest precision ratio, compared with other two methods, the DFT and the WT. It is because, in our method, the result from the DFT filtering is checked from the WT viewpoint.

5.3. Experiment II

In the experiment II, the three methods, DFT, WT and DFT+WT are compared from the recall ratio viewpoint, with change in the number of the database size. The result is shown in the figure 2.

As in the figure 2, the recall ratio of the DFT equals to the recall ratio of the DFT+WT. Both are superior to the recall ration of the WT. The comparison shows the validity of the DFT+WT, rather than WT+DFT.

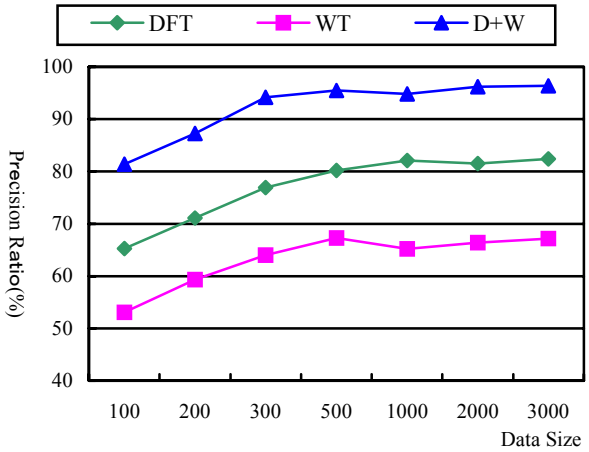

Figure 1. Result of Experiment I

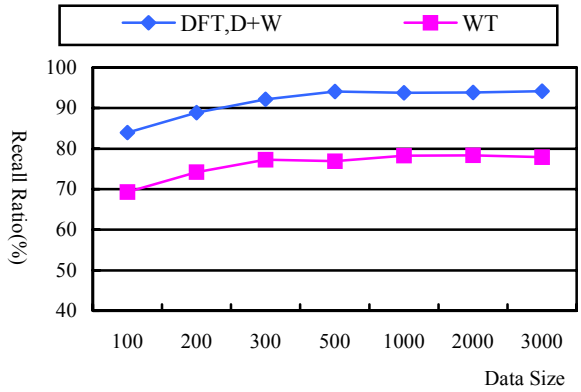

Figure 2. Result of Experimtent II

5.4. Experiment III

In the Experiment III, our method for the sub-matching search is evaluated. In the figure 3 shows the result of the precision ratio of the sub-sequence matching, with change in the number of the database size. The length of the sub-sequence given as a query varies as 1/2, 1/4 and 1/8 of the whole time period.

As shown in the figure 3, the precision ratio decreases depending on the length of the sub-sequence. In the sub-sequence matching, only a part of the sequence is alike as the given sub-sequence, which means the Euclid distance calculated for the overall sequence can be larger.

5.5. Experiment IV

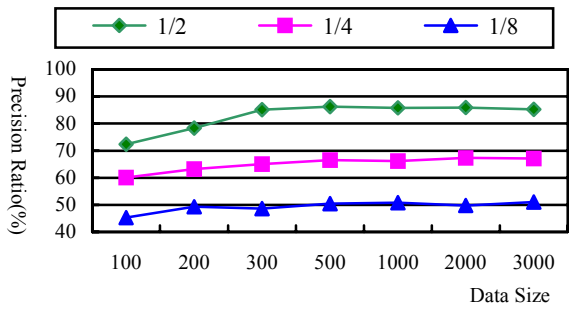

Figure 3. Result of Experiment III

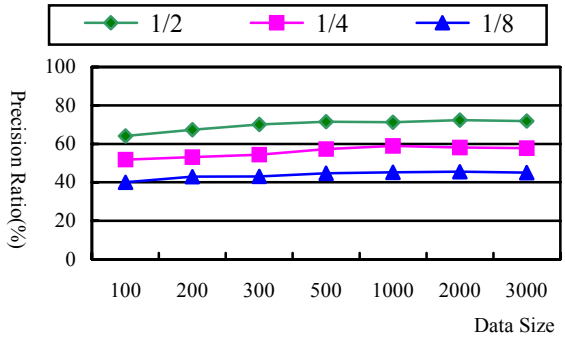

Figure 4. Result of Experiment IV

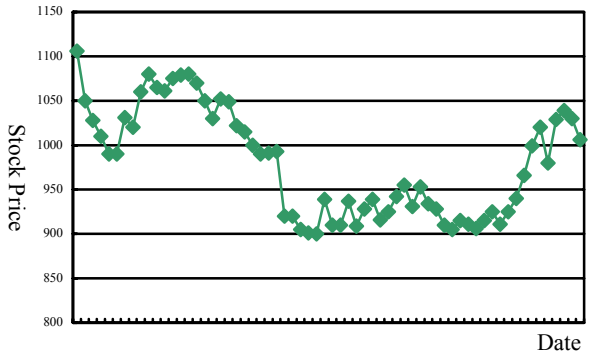

Figure 5. example of the time series data given as a query

In the experiment IV, our index size reduction method for the sub-matching search is evaluated. In the figure 3 shows the result of the recall ratio of the sub-sequence matching, with change in the number of the database size. The length of the sub-sequence given as a query varies as 1/2, 1/4 and 1/8 of the whole time period.

As the same as the case of the experiment III, the recall ratio decreases depending on the length of the sub-sequence. In the sub-sequence matching, only a part of the sequence is alike as the given sub-sequence, which means the Euclid distance calculated for the overall sequence can be larger.

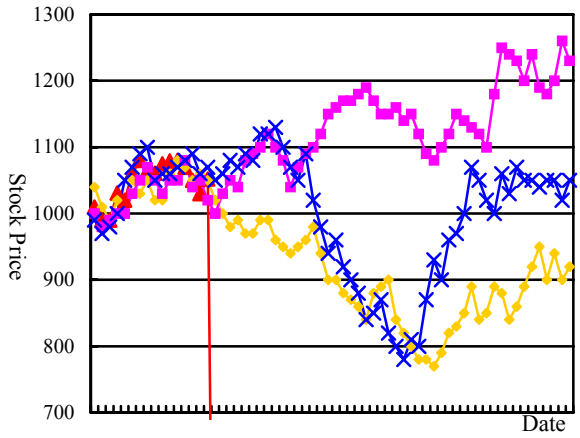

Figure 6. Samples of the reult (Length = 1/4)

The example of the sub-sequence used in the experiment III, is shown in the figure 5. In the figure 6, the samples of the result are shown.

5.6. Experiment V

In the experiment V, our index size reduction method for the sub-matching search is evaluated. In the figure 7 shows the result of the relationship between the threshold in clustering and the precision ratio.

As shown in the figure 7, as the less the threshold is, the less the precision ratio also decreases. Therefore, it is very important to select an appropriate Clustering Threshold with the proper precision ratio.

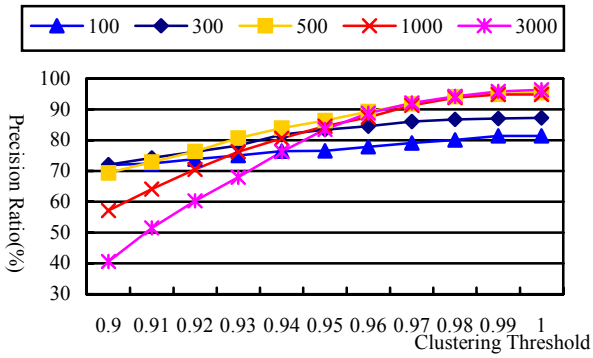

Figure 7. Result of Experiment V

5.7. Experiment VI

In the experiment VI, our index size reduction method for the sub-matching search is also evaluated from the recall ratio viewpoint. In the figure 8 shows the result of the relationship between the Clustering Threshold and the recall ratio.

Like in the experiment V, as shown in the figure 8, as the less the Clustering Threshold is, the less the precision ratio also decreases. Therefore, it is very important to select an appropriate Clustering Threshold with the proper recall ratio.

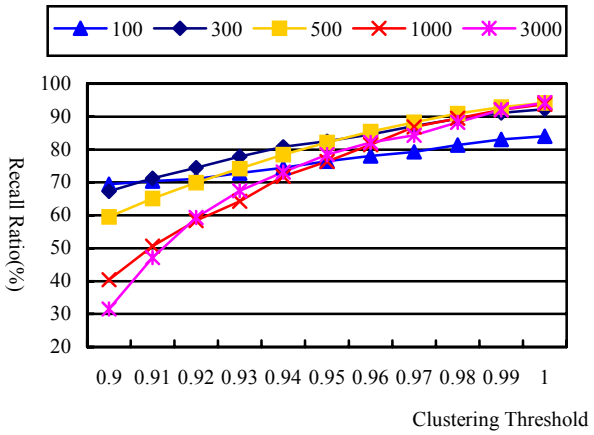

Figure 8. Result of Experiment VI

5.8. Experiment VII

In the experiment VII, our index size reduction method for the sub-matching search is evaluated again from the index size reduction viewpoint. In the figure 9 shows the result of the relationship between the Clustering Threshold in clustering and the index size reduction ratio.

Shown in the figure 9, depending on decreasing the Clustering Threshold, the index size can be reduced. From the figure 7, figure 8 and figure 9, the Clustering Threshold is appropriate to be set to a value from 0.94 to 0.96.

6. Several Applications

The method proposed in the paper is applied for the following applications:

- Management data
- Music data
- Video data

In the following, the overviews of these applications are described.

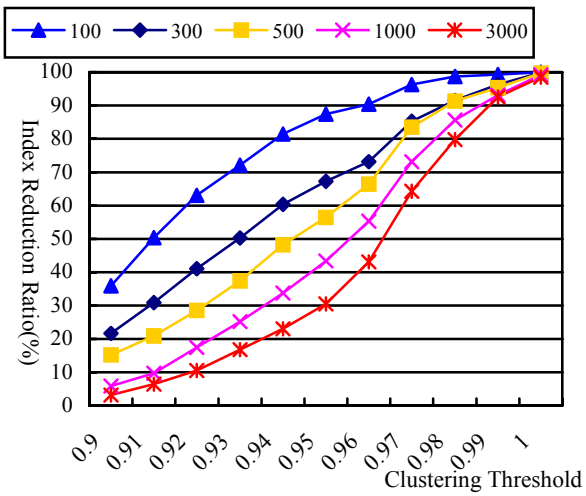

Figure 9. Result of Expeiment VII

6.1. Management Data

There are many corporate finance related time series data opened to the public, such as a capital a debt and a property. It is important and difficult to judge a status of a company from these data. In order to support the judgment, the similarity search can be applied with which some companies with similar sequence pattern can be chosen, given a sub-sequence of time-series data of the company in the critical status in the past. The problems are that each company has a lot of time-series data sequences. In order to reduce the number of the sequences and to be able to apply our proposed method, some management indices, such as earning ability ratio and payment ability ratio, are created using the stored finance data sequences. Then these indices, which are also time series data sequences, is combined into the single time series data, by put them in order. Finally, the obtained time series data can be searched using our method.

Some preliminary experiment has been made showing the precision ratio is 88% on average.

6.2. Music Data

Music data is time dependent data in nature. However, the method cannot directly be used for the similar music search because the number of the elements in the music sequence is extremely large. So, we basically transform the music MIDI data into two kinds of time series data sequences: sound level and tune. With these kinds of sequences, the similar music search can be easily realized. Compared with other characteristics such as feeling and music attributes such as a title and a composer, you can get more concise results. The simulation is in progress.

6.3. Video Data

Video data is similar to the music data, besides the video is a sequence of frames or images. The video data is not a time- series data. However, by transforming the video data into a set of time-series data sequences, our method can be applied. In order to do so, we calculate some characteristics for each frame data in a video data. Examples of such characteristics include average RGB values and brightness. These characteristics data compose some time series data sequences. With the proposed method, video subsequence matching can be easily realized. We observed from our preliminary simulation that the recall ratio is more than 80% and the precision ratio is about 20%.

7. Conclusion

In this paper, we presented a new method of similarity search method for time series data. The method proposed here contains two kinds of transforms, Discrete Fourier Transform (DFT) and Wavelet Transform (WT), used for filtering prior to checking Euclid distances. With this combination of the DFT and the WT, the search performance can be improved. From observation of out several experiments, our DFT+WT method is superior to both the DFT and the WT in the recall ratio and the precision ratio.

We also presented in the paper, that the index size can be reduced using a proposed method with the correlation coefficients, which is a problem in sub-sequence matching of time series data.

The proposed method can be applied for the existing time-series similarity search systems by employing the method as either a pre-filtering or a post-filtering. The number of similarity query result data can be reduced significantly though the index access cost will be increased a little. The authors are going to check the performance numbers by realizing the proposed method in some applications described in the section 6. Based on the proposed method, we are currently developing a prototype system for several new applications such as management data retrieval, music data retrieval and video data retrieval.

Acknowledgment

The authors would like to thank the referees of this paper for their helpful comments, and would also like to thank Yosuke Shoji, Hidetake Hase and Junya Fukumura for their valuable comments on the applications of the proposed method.

References

[1]K.P. Chan, A.W. Fu: Efficient Time Series Matching by Wavelets, ICDE, pp.126-133,1999.

[2]Christos Faloutsos: *Searching Multimedia Database by Content*, Kluwer Academic Publishers, 1996

[3]Struzik, Z. R. and Siebes A.: The Haar Wavelet Transform in the Time Series Similarity Paradigm, Procs. of Principles of Data Mining and Knowledge Discovery, pp.12-22, Sep,1999

[4]C. Sidney Burrus, R. A. Gopinath, and H. Guo : *Introduction to Wavelets and Wavelet Transforms*, A Primer ,Prentice Hall, 1997

[5]Amara Graps *:An introduction to wavelets*, IEEE, 1995

[6]Rakesh Agrawal, Christos Faloutsos, and Arun swami: Efficient similarity search in sequence databases , In Proc. of the Fourth International Conference on Foundations of Data Organization and Algorithms, pp.69-84, 1993.

[7]Christos Faloutsos, M. Ranganathan, and Y. Manolopoulos: Fast subsequence matching in time-series databases, In Proc. of the ACM SIGMOD Conference on Management of Data, pp. 419-429, 1994.

[8]Daniel Wu, D. Agrawal, A. E. Abbadi, A. Singh, and T. R. Smith: Efficient retrieval for browsing large image databases, In Proc. Conf on Information and Knowledge Management, pp.11-18, 1996.

[9]Flip Korn, H.V.Jagadish, and Christos Faloutsos: Efficiently supporting ad hoc queries in large datasets of time sequences, In Proc. of the ACM SIGMOD Conference on Management of Data, pp.289-300, 1997

[10]Norbert Beckmann, Hans-Peter Kriegel, Ralf Schneider, and Bernhard Seeger :The R*-tree, An efficient and robust access method for points and rectangles, In Proc. of ACM SIGMOD Conference on Management of Data, pp.322–330, 1990.

[11]Stefan Berchtold, Daniel A. Keim, and Hans-Peter Kriegel: An index structure for high-dimensional data, In Proc. of the 22nd VLDB Conference, pp.28-39, 1996.

[12]King Lum Cheung and A. Fu: Enhanced nearest neighbor search on the R*-tree, ACM SIGMOD Record, pp.16-21, Sept, 1998.

[13]Tzi cker Chiuen: Content-based image indexing, In Proc. of the 20th VLDB Conference, pp/582–593, 1994

[14]Antonin Guttman: R-trees, A dynamic index structure for spatial searching, In Proc. of the ACM SIGMOD Conference on Management of Data, pp.47–57, 1984

[15]Yi-Leh Wu, Divyakant Agrawal and Amr El Abbadi: A Comparison of DFT and DWT Based Similarity Search in Time-Series Databases, CIKM2000, pp.488-495, 2000

[16]C. Faloutsos and K.I. Lin FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets ACM SIGMOD 95, pp. 163-174,1995

[17]Piotr Indyk, N. Koudas and S. Muthukrishnan: Identifying Representative Trends in Massive Time Series Datasets Using Sketches. In the 26th International Conference on Very Large Databases (VLDB), 2000