A New Approach for Similarity Search in Time Series Databases Based on Slopes

Durga Toshniwal
Department of Electronics and Computer Engineering
Indian Institute of Technology
Roorkee , Uttaranchal – 247 667
India
durgadec@iitr.ernet.in

R. C. Joshi
Department of Electronics and Computer Engineering
Indian Institute of Technology
Roorkee , Uttaranchal – 247 667
India
joshifcc@iitr.ernet.in

Abstract – **The problem of similarity search in time series databases has gained great significance in the recent times. In this paper we introduce a new and simple approach for performing similarity search over time series data. This technique is based on the observation that similar time sequences will have similar variations in their slopes. The proposed technique is capable of handling queries of variable lengths and works irrespective of global scaling or shrinking of time sequences. It is also capable of handling vertical shifts.**

I. INTRODUCTION

Time series constitute a large portion of data stored in computers. Some typical examples include stock prices, biomedical data, atmospheric data, and so on. In last decade, there have been several attempts to model time series data, to design languages to query such data, and to develop access structures to efficiently process queries on such data. The problem of similarity search in time series data is important and non-trivial.

In order to perform similarity search on time series data, we need indexing methods that are capable of supporting efficient retrieval and matching of time series data. Most of the indexing methods for multi-dimensional data such as the R-tree [1] and the R*-tree [2] degrade performance at dimensionalities greater than 8-10 [3] and eventually perform almost like sequential scanning algorithms at high dimensionalities. Thus, to utilize multi-dimensional indexing techniques, it is essential to first perform dimension reduction on time series data. Dimension reduction maps high-dimensional data to a lower dimension space. Next, some distance measure such as the Euclidean Distance may be used to calculate the distance and hence the similarity between any two time sequences.

There are many ways to perform dimension reduction. Some of the commonly used methods for performing dimension reduction include Discrete Fourier Transform (DFT) [4, 5, 6, 7], Discrete Wavelet Transform (DWT) [8, 9, 10, 11, 12], Singular Value Decomposition (SVD) [13] and Piecewise Aggregate Approximation (PAA) [14].

The DFT is very well suited for naturally occurring signals which are sinusoidal in nature but it is ill-suited for representing signals having discontinuities.

The Haar is the most commonly used Wavelet Transform used for dimension reduction. As the basis function for Haar is not smooth, the Haar Wavelet Transform approximates any signal by a ladder like structure. Thus the Haar Wavelet Transform is not likely to approximate a smooth function using only a few coefficients. So the number of coefficients to be added must be high. Finding wavelets having more continuous derivatives is still an active area of research.

The SVD technique uses the KL transform for performing dimension reduction. The key weakness of this approach is that the SVD is data dependent. This means that it uses the dataset to determine new basis vectors. So it has to be recomputed whenever a database item is updated. Thus, the recomputation time becomes infeasible for practical purposes especially when the database is very large.

In case of PAA, the time sequence is divided into equal length segments. The corresponding feature sequence comprises mean values of each segment. But the means representing each segment give only a rough approximation of each time sequence.

Most of the approaches for performing similarity search in time series data developed so far rely on dimension reduction. This may lead to loss of information of some kind.

In this paper, we introduce a novel technique for similarity search in time series databases. It is based on the assumption that similar time sequences will have similar variations in their slopes. The proposed technique is simple and capable of handling variable length queries on time series data. It is also capable of handling different scaling factors and baselines. Moreover there is no need to perform any kind of data compression by means of dimension reduction. The performance of the proposed technique is independent of the number of datapoints in the candidate or query time sequences.

The rest of the paper is organized as follows. Section II gives related work. Section III describes the proposed approach. In Section IV, we give a case study and finally conclusions and directions for future work are covered in Section V.

II. RELATED WORK

In this section we briefly discuss some key approaches for performing similarity search in time series data based on dimension reduction.

Agrawal et al. [4] used the Discrete Fourier Transform to perform dimension reduction. The DFT was used to map the time sequences to the frequency domain and the index so built was called the F-index. For most sequences of practical interest, the low frequency coefficients are strong. Thus the first few Fourier coefficients are used to represent the time sequence in frequency domain. These coefficients were indexed using the R*-tree [2] for fast retrieval. The basis for this indexing technique is Parseval's theorem. The Parseval's theorem guarantees that the distance between two sequences in the frequency domain is the same as the distance between them in the time domain. For a range query the F-index returns a set of sequences that are at a Euclidean Distance ∈ from the query sequence.

The F-index may raise false alarms but does not introduce false dismissals. The actual matches are obtained in a post-processing step wherein the distance between the sequences are calculated in the time domain and those sequences which are within ∈ distance are retained and the others are dismissed. The F-index typically handles 'whole matching' queries.

Faloutsos et al. generalized the F-index method in [15] and called it the ST-index. In this technique, subsequence queries are handled by mapping data sequences into a small set of multidimensional rectangles in feature space. These rectangles are indexed using spatial access methods like the R*-tree [2].

A sliding window is used to extract features from the data sequence resulting in a trail in the feature space. These trails are divided into sub-trails which can be represented by their Minimum Bounding Rectangles (MBR). Thus, in place of storing all the points in a trail, only a few MBRs are stored. When a query is presented to the database, all the MBRs intersecting the query region are retrieved. This guarantees no false dismissals but also raises some false alarms as sub-trails that do not intersect the query region but their MBRs are also retrieved.

Chan et al. [8] have proposed to use the DWT in place of DFT for performing dimension reduction in time series data. Unlike the DFT which misses the time localization of sequences, the DWT allows time as well as frequency localization concurrently. The DWT thus bears more information of signals in contrast to DFT in which only frequencies are considered. The approach in [8] employed the Haar Wavelet Transform for mapping high-dimensional time series data to lower dimensions.

A data dependent indexing scheme was proposed in [13] and is known as the SVD method for dimension reduction. The database consists of n-dimensional points. We map them on a k-dimensional subspace, where $k < n$, maximizing the variations in the chosen dimensions. An important drawback of this approach is the deterioration of performance upon incremental update of the index. Therefore the new projection matrix should be calculated and the index tree has to be reorganized periodically to keep up the search performance.

In PAA [14], each time sequence say of length k is segmented into m equal length segments such that m is a multiple of k. If that is not the case, then the sequence is padded with zeros in order to perform the segmentation. The averages of segments together form the new feature vector for the sequence. The correct selection of m is very important because if m is very large, the approximation becomes very rough but if m is very small, the performance deteriorates.

A. Euclidean Distance

Mostly similarity search methods utilize the Euclidean distance model for calculating the similarity between the query and candidate sequence. According to this model, if the Euclidean Distance $D\ (X,\ Y)$ between two time sequences X and Y of length n is less than a threshold ∈, then the two sequences are said to be similar.

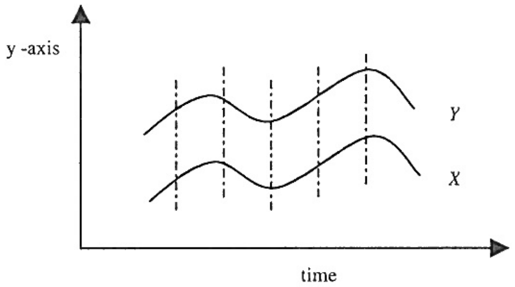

Fig.1. Two similar time sequences with vertical shifts between them

Thus:

$$D\,(X,\,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \qquad (1)$$

There is a major shortcoming in the Euclidean distance model. According to Fig. 1, X and Y are similar to each other as Y is obtained just by vertically shifting X. But when X and Y are compared using (1), they may be evaluated as dissimilar due to the vertical shifts existing between the two at each i. When compared by our proposed approach, X and Y will be interpreted as similar even though a vertical shift exists between them. The reason being that they have similar cumulative variations in slopes.

III. THE PROPOSED APPROACH

We propose to use the cumulative variation in slopes for performing similarity search in time series data. In this paper, we assume that a time series consists of a sequence of real numbers which represent the values of a measured parameter at equal intervals of time. Let the time series database consist of p time sequences designated by X_1, $X_2 \dots X_p$. Each time sequence X_i in turn can be represented as $< (t_{i1},\, y_{i1}),\, (t_{i2},\, y_{i2}) \dots (t_{in},\, y_{in}) >$ where n is the number of samples in the time sequence.

In the proposed approach, each of the candidates X_i in the time series database is first scaled along the time axis so that their time axes become equal to some desired time t_d. The selection of t_d is done by the user and may depend on the domain of application of the data. In our technique, scaling along the time axis is done to help compare variable length time sequences. For example, a 5-year sales pattern of a Company A can be compared to a 10-year sales pattern of Company B. Another example where scaling can play a crucial role is the comparison of the growth of a tumour for the past 10-months versus the growth of the tumour for past 10-days. In order to avoid any distortions that may arise due to time scaling, the values along the y-axis for each X_i are also scaled proportionately. Thus each transformed X_i denoted by X_i' may be represented as $< (t_{i1}',\, y_{i1}'),\, (t_{i2}',\, y_{i2}') \dots (t_{in}',\, y_{in}') >$ where:

$$t_{ij}' = t_{ij} * (\, t_d / t_{in}\,)$$
$$\text{and } y_{ij}' = y_{ij} * (\, t_d / t_{in}\,) \qquad (2)$$

In our approach, we have considered variations in the y-values of each X_i about the mean as the range of y-values may vary substantially. Thus, each y_{ij}' is divided by the corresponding mean for y-values y_m to obtain y_{ij}'' where:

$$y_m = (y_{i1}' + y_{i2}' + \ldots + , y_{in}') / n \tag{3}$$

This is followed by dividing each of the time sequences in the database into same number of small, equi-width strips along the time-axis.

The same procedure is repeated for any query Q for similarity search. Or in other words, the query is first time scaled to t_d and then scaled proportionately along the y-axis followed by dividing the y-values by their mean. The resulting sequence is divided into small, equi-width strips.

Thus, we consider both the query and the candidate to be comprised of same number of small strips along the time axis .The strips have different heights but same widths along the time-axis as shown in Fig 2.

Finally, we compute the parameter for variations of slopes between any two sequences Q and C as:

$$S\,(Q, C) = \sqrt{\sum (S_{qj} - S_{cj})^2} \tag{4}$$

where S_{cj} and S_{qj} are the slopes for the j^{th} strip in the candidate time sequence C (in the present case X_i) and the query time sequence Q respectively :

$$S_{cj} = \{ y_{tc}''{}_{(j+1)} - y_{icj}'' \} / \Delta t \tag{5}$$

and $$S_{qj} = \{ y_q''{}_{(j+1)} - y_{qj}'' \} / \Delta t \tag{6}$$

We assume in (5) and (6) that the starting and ending coordinates for the j^{th} strip of the candidate are given by (t'_{icj}, y_{tcj}'')and ($t'_{ic(j+1)}$, $y_{ic}''{}_{(j+1)}$). Similarly, the starting and ending coordinates for the j^{th} strip of the query time sequence are given by (t'_{qj}, y_{qj}'') and ($t'_{q(j+1)}$, $y_q''{}_{(j+1)}$). And Δt is the width of each of the strips and is a constant. The choice of Δt may be user specified or domain specific. The important thing to note about the selection of Δt is that its value should be optimally selected so that it is neither too small (because that may lead to excessive computations) nor too large (loss of details).

Ideally for two exactly similar time sequences, the value of the parameter $S\,(Q, C)$ must be zero. Practically, the smaller the value of $S\,(Q, C)$, the more is the similarity between the time sequences under comparison. For range queries and nearest-neighbour queries we may choose to have $S\,(Q, C) \leq l$ where l specifies some degree of tolerance allowed while performing similarity search in the time-series database. The overall strategy thus involves the following steps:

Step 1: Scaling of data along the time-axis to allow variable length queries.

Step 2: Correspondingly scaling the values of y-ordinates to avoid any possibility of data distortions.

Step 3: Dividing each value along the y-axis by the mean of the y-values.

Step 4: Dividing each time sequence into same number of small, equi-width strips.

Step 5: Computing the parameter $S\,(Q, C)$ for variations in slopes of the two time sequences under comparison. Ideally, it should be zero.

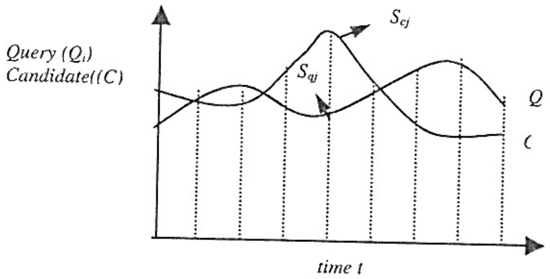

Fig.2. Query and candidate time sequences divided into strips

IV. PERFORMANCE EVALUATION

We have evaluated the performance of the proposed technique by considering synthetic sample time sequences as the test data.

The first set of sample sequences *H1*, *H2* and *H1-Reverse* are shown in Fig. 3. The scaled data is shown in Fig. 4. After the proposed technique is applied to them (t_d = 2.0), the resultant transformed time sequences are designated by *H1t, H2t and H1-Reverset* are shown Fig. 5. This is followed by the computation of parameter for variation in slopes as in (4) between the query *(H1t)* and the candidate time sequences *(H2t, H1-Reverset)*. We have also compared our results with the commonly used similarity model based on Euclidean distance as in (1).

Table I clearly indicates the inability of the Euclidean distance model to identify similar time sequences which are globally expanded versions of each other. As shown in Fig. 3, sequence pair *H1* and *H2* are such an example. It can be seen from Table I that the Euclidean distance between *H1t* and *H2t* is less than that between *H1t* and *H1-Reverset.* On the contrary, the proposed technique shows that the variation in slopes of *H1t* and *H1- Reverset* is about 6 times that between *H1t* and *H2t*. The greater the variation, the more is the dissimilarity. Thus, in turn, as per our approach, *H1* and *H2* are very similar as compared to *H1* and *H1-Reverse* as can also be seen in Fig. 3.

Similarly, sequences *V1*, *V2* and *V1'* are shown in Fig. 6. After applying our approach, finally transformed *V1, V2* and *V1'* are designated by *V1t, V2t* and *V1't* and are shown in Fig. 7. Taking *V1t* as the query sequence in this case, the results are shown in Table II. In this case, the results as indicated by parameter *S* as well as by the Euclidean distance approach are parallel. Thus it can be concluded that *V1* and *V2* are very dissimilar to each other in contrast to *V1* and *V1'* which are quite similar to each other.

Fig. 8 shows another set - *V3, V4* and *V5* where *V3* is taken as the query and *V4, V5* are candidate time sequences. The transformed sequences are shown in Fig 9 and are designated by *V3t, V4t* and *V5t*. The results of the proposed approach and the Euclidean distance model are shown in Table III. As expected, the variations in the slopes *S* between *V3t* and *V4t* is almost 200% that of the variations in slopes between *V3t* and *V5t.* So we conclude that *V3* and *V5* are similar to each other and on the contrary *V3* and *V4* are very dissimilar. The results of the Euclidean distance model also indicate the same.

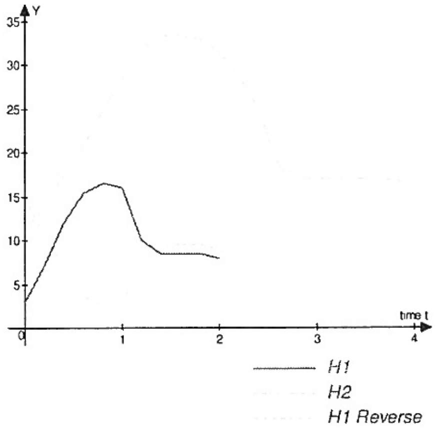

Fig.3. The sequence *H1* is taken as the query and *H2* and *H1- Reverse* are the candidate time sequences

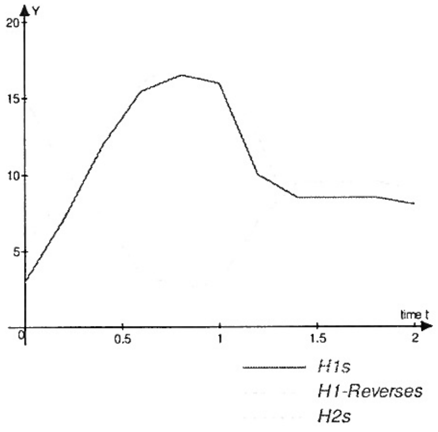

Fig.4. Scaled sequences designated by *H1s* , *H2s* and *H1- Reverses*

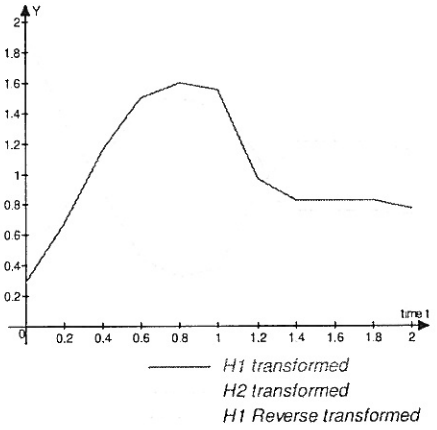

Fig.5. Finally transformed sequences *H1, H2,* and *H1-Reverse* designated by *H1t, H2t,* and *H1-Reverset*

TABLE I

PARAMETER FOR VARIATIONS IN SLOPES *S (Q, C)* VERSUS EUCLIDEAN DISTANCE D *(Q, C)*

Sequence Pairs	Parameter *S*	Euclidean Distance D
H1t, H2t	1.19	2.81
H1t, H1-Reverset	9.86	1.66

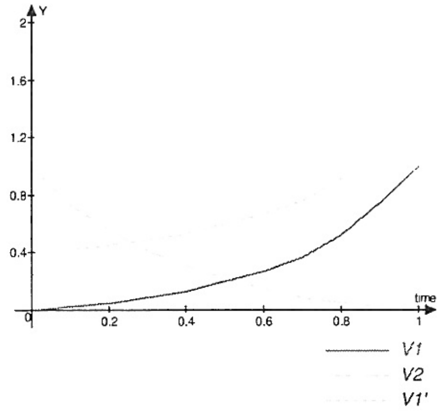

Fig.6. The sequence *V1* is taken as the query and *V2* and *V1'* as the candidate time sequences

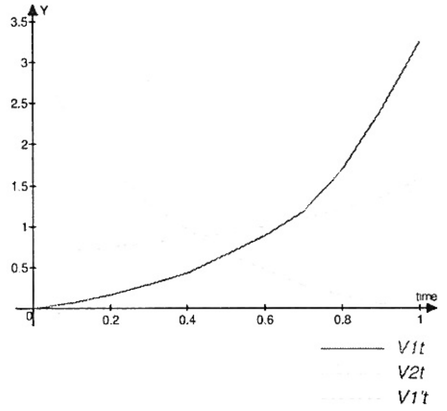

Fig.7. Finally transformed sequences *V1, V2,* and *V1'* designated by *V1t, V2t,* and *V1't*

TABLE II

PARAMETER FOR VARIATIONS IN SLOPES *S (Q, C)* VERSUS *D (Q, C)*

Sequence Pairs	Parameter *S*	Euclidean Distance *D*
V1t, V2t	20.83	6.14
V1t, V1't	3.37	2.36

We have also considered *N1, N2* and *N3* (Fig. 10) as the next sample data time sequences where *N1* is taken as the query and *N2* and *N3* are taken as the candidate sequences. The transformed sequences are shown in Fig. 11. It can be clearly seen from Table IV that *N1t* and *N2t* are very similar to each other as compared to *N1t* and *N3t*. The Euclidean distance computations also indicate the same.

Fig. 12 shows synthetically generated random sample data. The results of the proposed approach have been shown in Table V. Fig. 13 shows the transformed sequences.

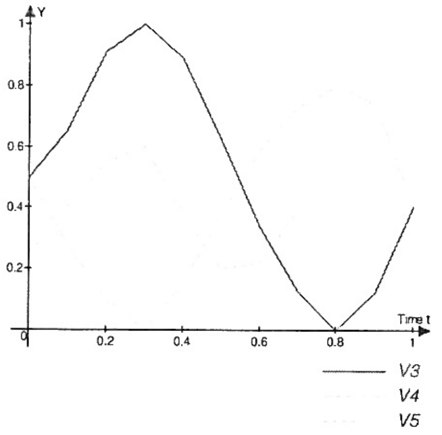

Fig.8. The sequence *V3* is taken as the query and *V4* and *V5* are the candidate time sequences

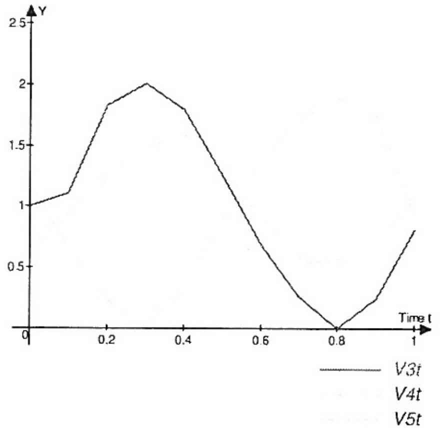

Fig.9. Finally transformed sequences designated by *V3t*, *V4t*, and *V5t*

TABLE III

PARAMETER FOR VARIATIONS IN SLOPES *S (Q, C)* VERSUS *D (Q, C)*

Sequence Pairs	Parameter *S*	Euclidean Distance *D*
V3t, V4t	26.93	4.18
V3t, V5t	12.28	1.99

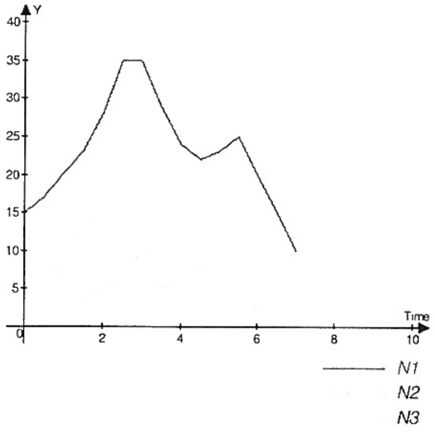

Fig.10. The sequence *N1* is taken as the query and *N2, N3* are taken as the candidate time sequences

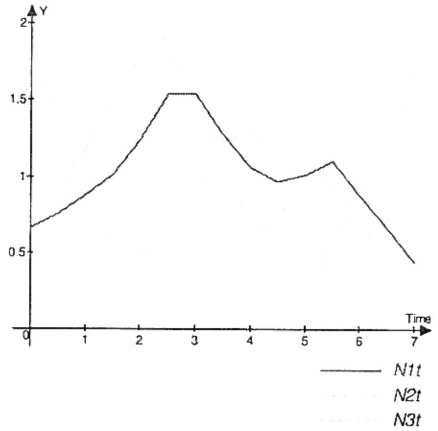

Fig.11. Finally transformed sequences designated by *N1t*, *N2t*, and *N3t*

TABLE IV

PARAMETER FOR VARIATIONS IN SLOPES *S (Q, C)* VERSUS *D (Q, C)*

Sequence Pairs	Parameter *S*	Euclidean Distance *D*
N1t, N2t	2.34	1.69
N1t, N3t	3.07	2.62

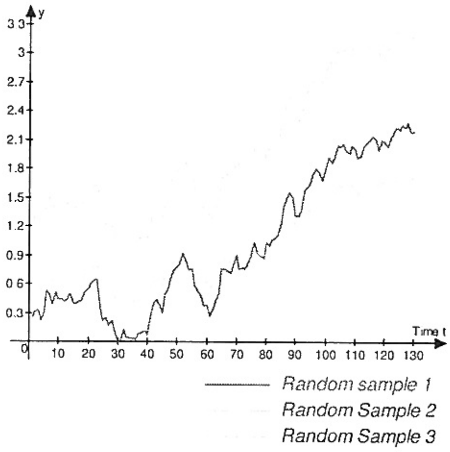

Fig.12. The sequence *Sample 1* is taken as the query and *Samples* 2 and *3* are taken as the candidate time sequences

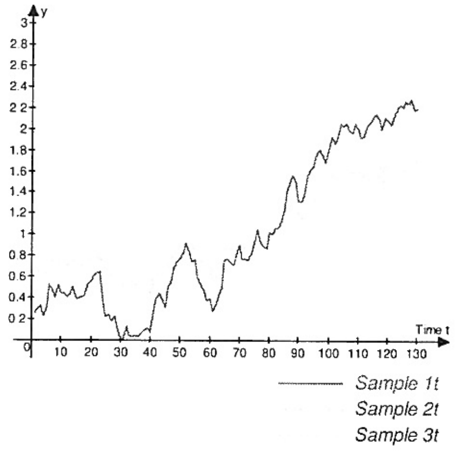

Fig.13. Finally transformed sequences

TABLE V

PARAMETER FOR VARIATIONS IN SLOPES *S (Q, C)* VERSUS *D (Q, C)*

Sequence Pairs	Parameter *S*	Euclidean Distance *D*
Sample 1t, Sample2t	0.39	3.28
Sample 1t, Sample3t	0.49	4.12

V. CONCLUSIONS AND FUTURE WORK

In this paper, a simple and effective technique for performing similarity search in time series data has been proposed. The given time sequences are time scaled and brought to the same time range. The y-values are also proportionately scaled by the same factor and only the variations in the y-values about the mean are retained. This way the given time sequences are transformed. The computation of the parameter for variations in slopes is done on the transformed data. Experiments show that the proposed technique can handle vertical shifts in the time sequence data, global scaling of the data and allows variable length queries. The proposed approach does not involve any dimension reduction and hence the data distortions arising out of it are avoided. Euclidean distance model has also been used to compare the test data considered. Our approach provides quantitatively better comparisons.

In this approach we have assumed that a time series comprises of samples of a single measured variable against time. In future work, we intend to broaden its scope so that it can handle multivariable time sequences. We propose to further refine our technique by assigning weights to locations of the slopes along the time axis. We also intend to develop alternate parameters for assessing similarity in time series data, which may be used individually, or in conjunction with each other.

VI. REFERENCES

[1] A. Guttman, "R-trees: A dynamic index structure for spatial searching," in *Proceedings of the 1984 ACM SIGMOD Conference,* pp. 47-57.

[2] Norbert Beckmann, Hans-Peter Kriegel, Ralf Schneider, and Bernhard Seeger, "The R*-tree: An efficient and robust access method for points and rectangles," in *Proceedings of 1990 l ACM SIGMOD Conference,* pp. 322-331.

[3] K.V. Kanth, D. Agrawal, and A. Singh, "Dimensionality reduction for similarity searching in dynamic databases," in *Proceedings of the 1998 ACM SIGMOD Conference*, pp. 166-176.

[4] R. Agrawal, C. Faloutsos, and A. Swami, "Efficient similarity search in sequence databases," in *Proceedings of the 1993 4th International Conference on Foundations of Data Organization and Algorithms,* pp. 69-84.

[5] K. Chu and M. Wong, "Fast time-series searching with scaling and shifting," in *Proceedings of the 1999 18th ACM Symposium on Principles of Database Systems,* pp. 237-248.

[6] C.Faloutsos, H. Jagadish, A. Mendelzon, and T. Milo, "A signature technique for similarity based queries," in *Proceedings of the 1997 International Conference on Compression and Complexity of Sequences.*

[7] D. Refiei, "On similarity based queries for time series data," in *Proceedings of the 1999 15th IEEE International Conference on Data Engineering*, pp. 410-417.

[8] K. Chan and A. W. Fu, "Efficient time series matching by wavelets," in *Proceedings of the 1999 15th IEEE International Conference on Data Engineering,* pp. 126-133.

[9] Y. Wu, D. Agrawal, and A. El Abbadi, "A comparison of DFT and DWT based similarity search in time series databases," in *Proceedings of 2000 9th ACM International Conference on Information and Knowledge Management,* pp. 488-495.

[10] T. Kahveei and A. Singh, "Variable length queries for time series data," in *Proceedings of 2001 17th International Conference on Data Engineering*, pp. 273-282.

[11] Z. Struzik and A. Siebes, "The haar wavelet transform in the time series similarity paradigm," in *Proceedings of 1999 Conference on Principles of Data Mining and Knowledge Discovery,* pp. 12-22.

[12] C. Wang and X. S. Wang, "Supporting content based searches on time series via approximation," in *Proceedings of 2000 International Conference on Scientific and Statistical Database Management,* pp. 69-81.

[13] F. Korn, H. Jagadish, and C. Faloutsos, "Efficiently supporting ad hoc queries in large datasets of time sequences," in *Proceedings of 1997 ACM SIGMOD International Conference on Management of Data,* pp. 289-300.

[14] Byoung-Kee Yi and C. Faloutsos, "Fast time sequence indexing for arbitrary Lp norms," *The VLDB Journal,* 2000, pp. 385-394.

[15] C.Faloutsos, M. Ranganathan, and Y. Mano Lopoulos, "Fast subsequence matching in time-series databases," in *Proceedings of 1994 ACM SIGMOD International Conference on Management of Data,* pp. 419-429.