Efficient LCSS Distance Measure for Searching of Similar Time Series Trajectories

Gajanan S. Gawde
Department of Computer Engineering
Goa College of Engineering Farmagudi – Ponda, Goa – India

Jyoti D. Pawar
Department of Computer Science & Technology
Goa University Taleigao-Panaji
Goa-India

Abstract

Many researchers have been attracted towards searching of similar moving objects trajectories due to its wide range of real time applications. Searching of similar trajectories of moving objects helps data mining users to take smart decisions and thereby improving the performance of systems. Trajectories are compared for similarity using edit distance measures such as DTW, ERP, EDR, and LCSS. These existing distance measures are popular distance measures and compare trajectories for similarity by computing proximity distance between them. Distance measures DTW, EDR, ERP, and LCSS support scaling and translation property but it does not support rotation invariant property. RI distance measure supports scaling, translation and rotation invariant property and hence RI distance measure is considered to be superior compared to other edit distance measures. Even though RI distance measure is better compared to other edit distance measures, it has two main drawbacks. The first one is, it does not compares trajectories based on the shape and this shape based searching is very much required, since proximity distance is not only the best way to compares trajectories. The second problem, RI is not robust to noise and produces poor results. In this paper, we have proposed Efficient Longest Common Sub-Sequence (ELCSS) distance measure to compares trajectories based on the shape feature. ELCSS distance measure is based on the angular distance of the trajectories. The angular distance captures the shape feature of the trajectory. We have carried out experimental study on the real time and synthetic datasets. Experimental results reveal that our proposed ELCSS distance measure compares the trajectories based on the shape feature. Further, our experimental results reveal that, ELCSS distance measure supports rotation invariant property and very robust to the noise.

Keywords: Efficient Longest Common Sub-Sequence Distance Measure, Time Series Trajectories, Shape Based Matching, Turning Function

I. INTRODUCTION

Moving object trajectory is a continuous line segments obtained by joining sequence of points with change in the time. Trajectory is a path representing the movement of moving object as the time changes. Trajectory can be represented in two or three dimensional space. In case of two dimensional space (x,y) parameters are required where x is latitude and y is longitude. In three dimensional space (x,y,z) parameters are required where x and y are same as two dimensional and z is altitude. General form of Trajectory representation is T(oid, t,x,y,z) where T is name of trajectory, oid is moving object, t is the time at which object position was recorded and (x,y,z) represent the position of the moving object at time t. Time series trajectories are gaining lot of importance due to wide range of applications.

DTW, ERP, EDR and LCSS distance measures are based on the edit distance and these measures are considered to be popular distance measures. These existing edit distance measures are invariant of scaling and translation but not invariant of rotation. Edit distance measures are comparing time series trajectories based on the proximity distance and not based on the shape feature. There is a need to compares time series trajectories based on the shape feature, in order to improve the performance of comparisons. We have provided two motivational examples to discuss the drawback faced by edit distance measures.

A. Example 1

Consider the trajectories of synthetic dataset as shown in figure 1. Query trajectory Q is compared with the three trajectories for similarity using DTW, ERP, EDR and LCSS measures. The result of comparison of trajectories is shown in table 1. Query trajectory Q is similar to the three trajectories T1, T2 and T3. DWT, ERP, EDR and LCSS distance measures are not able to show proper match of the trajectories. This is due to edit distance measures are not comparing trajectories based on the shape feature rather it is simple proximity distance. And also these distance measures are not rotation invariant.

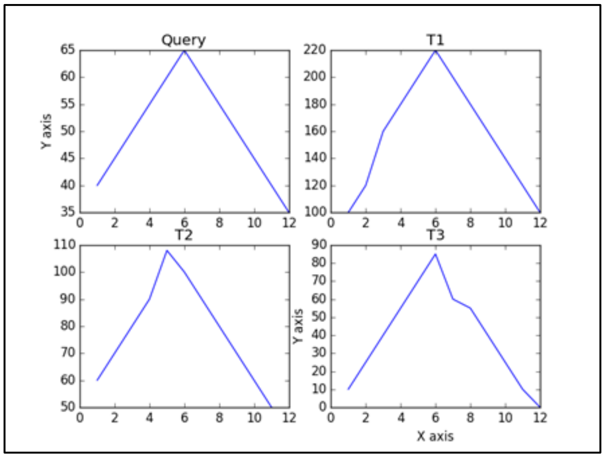

Fig. 1: Synthetic Trajectories Dataset 1

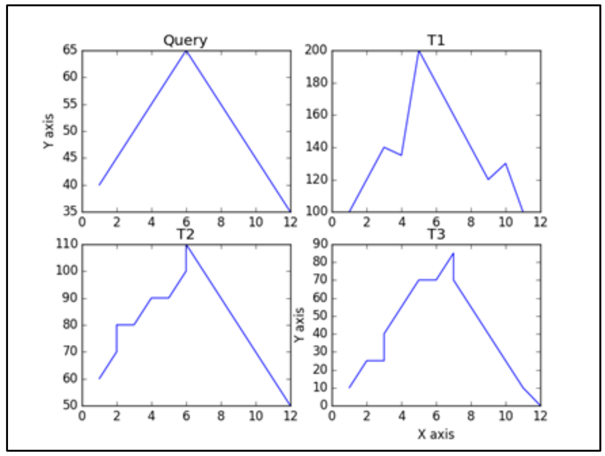

Fig. 2: Synthetic Trajectories Dataset 2

Table - 1
Similarity Distance on Dataset 1

Distance Measure	*(Q,T1)*	*(Q,T2)*	*(Q,T3)*
DTW	*328*	*324*	*361*
ERP	*228*	*207*	*241*
EDR	*4*	*4*	*4*
LCSS	*0*	*0*	*0*

Table - 2
Similarity Distance on Dataset 2

Distance Measure	*(Q,T1)*	*(Q,T2)*	*(Q,T3)*
DTW	*328*	*324*	*361*
ERP	*228*	*207*	*241*
EDR	*4*	*4*	*4*
LCSS	*0*	*0*	*0*

B. Example 2

Consider the synthetic dataset 2 of trajectories as shown in figure 2. This synthetic dataset was generated with noise in the trajectories. The query trajectory Q is compared with T1, T2 and T3 trajectories for similarity and result is as shown in table 2. Query trajectory is similar to T1, T2 and T3 trajectories. DWT, ERP, EDR and LCSS distance measures are not able to show proper match of the trajectories. This is due to edit distance measures are not robust to the noise. Due to this noise in the dataset, the edit distance measures are generating very poor results.

We conclude following points from Example 1 and 2 respectively. In Example 1, the query trajectory is a proper match with the three trajectories but DTW, ERP, EDR and LCSS distance measures fail to identify as proper match. This is due to fact that, DTW, ERP, EDR and LCSS measures are not comparing trajectories based on shape feature. Also, these edit distance measures are not rotation invariant. So, there is a need to compares trajectories based on shape feature with rotation invariant property.

In Example 2, the query trajectory is similar to the three trajectories from the dataset. This dataset was generated with noise, so that robustness of the edit distance measures are tested. DTW, ERP, EDR and LCSS distance measures fail to identify as proper match. There is a need of a strong distance measure which will be robust to a noise.

In this paper we have proposed Turning Function Based Distance measure in order to tackle the problems faced by DTW, ERP, EDR and LCSS distance measures. Turning Based Distance measure compares trajectories based on shape feature and it is robust to a noise.

Contribution of our work in this paper is as follows:

1) We have proposed ELCSS distance measure to compares trajectories based on the shape feature.
2) ELCSS distance measure supports rotation invariant property.
3) ELCSS distance measure is robust to a noise.

The rest of paper is organized as follows. In section 2 we present related work. In section 3, we have proposed ELCSS Distance measure for comparing trajectories for similarity. Section 4, we have discussed ELCSS algorithm. Section 5, present the experimental results of the proposed ELCSS techniques. Finally, section 6 concludes the paper.

II. RELATED WORK

Searching of similar time series trajectories is quite interesting problem and quite a few researchers have contributed towards this field. The pioneering work by (1) , (2) compare trajectories for similarity using Euclidean distance measure. The time series are scaled and transformed efficiently and compared for similarity in (3). (4) Introduced DTW to allow a time series trajectories to be stretched to provide better match with the trajectories. The performance of DTW was highlighted in (5) and subsequently enhanced using lower bounding distance measure with segmentation.

In (6) and (7), LCSS measure is used to compares trajectories for similarity. LCSS measure allows a variable length gap to be inserted during matching of the trajectories and hence robust to a noise. In (8) extended LCSS measure is defined to compares trajectories of the objects from video. First 3D trajectories are extracted from a video and then LCSS distance measure is used to compares trajectories. Searching of similar multi-dimensional trajectories are explored in (7). Authors have used LCSS distance measure to compares muti dimensional trajectories. LCSS distance measure is used in conjection with sigmoidal function in (9) to compare trajectories for similarity. In (10), indexed multi-dimensional trajectories supporting multiple distance measures such as LCSS, DTW are proposed and index structure was design in a such a way that there was no need to rebuild index again and again. Extracting trajectories from videos have been explored by few researchers. In (8), modified LCSS distance measure is used to compares the trajectories for similarity. (11) proposed new distance measure using gaborfilter to measure the distance between the trajectories extracted from videos. Dynamic Time Waring distance measure is used in (12) to compares the trajectories.

III. EFFICIENT LCSS DISTANCE MEASURE FOR TIME SERIES TRAJECTORIES

A. Basic Terminology Used

- Definition 1: Segment is defined as the line drawn between two points. Let P1 and P2 be the two points, then segment is a line joining two points P1 and P2.
- Definition 2: Time Series Trajectory of a moving object is defined as sequence of 2-Dimensional vectors, each describing the position of object at time instants 1 to M. Let R and S be the two trajectories with length M and N respectively and are described in 2-Dimensional space as

Follows

$$R = [(t_1, r_1), (t_2, r_2), (t_3, r_3), \ldots, (t_M, r_M)] \quad (1)$$

$$S = [(t_1, s_1), (t_2, s_2), (t_3, s3), \ldots, (t_M, s_M)] \quad (2)$$

- Definition 3: Rest function is defined as a function which return all points of trajectory except the first point.

$$R = [(t_1, r_1), (t_2, r_2), (t_3, r3), \ldots, (t_M, r_M)] \quad (3)$$

$$Rest(R) = [((t_2, r_2), (t_3, r3), \ldots, (t_M, r_M)] \quad (4)$$

- Definition 4: Turning Function Turning Function [ϴ(s)] is defined as the cumulative angle function, which gives the angle between the counterclockwise tangent and the x -axis as a function of the arc length s. ϴ(s) keeps track of the turning that takes place, increasing with left hand turns, and decreasing with right hand turns. Figure 3 shows Turning Function of a time series trajectory. The angle between two segments is computed and stored in the turning function. Initially, reference line is drawn and first angle is computed with respect to the reference line. T1 is the first angle with ref line as shown in the figure. Every time, tangent is drawn with the segment and the angle between tangent and segment is computed. T1 to T13 are the angles of the trajectory in the turning function representation.

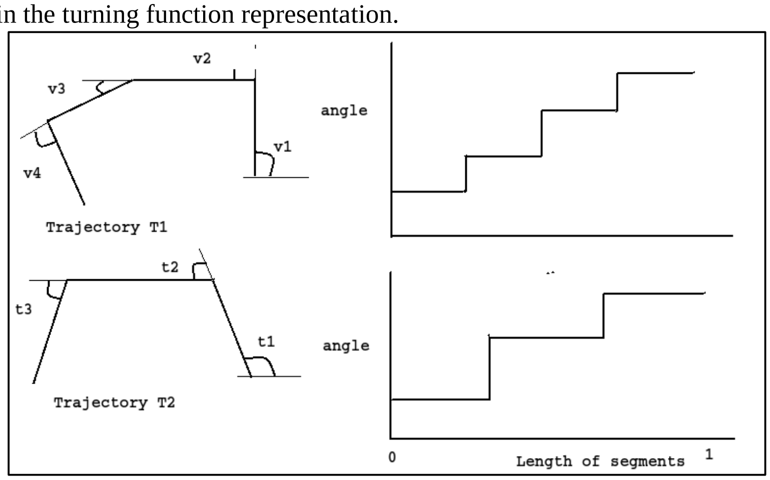

Fig. 3: Turning Function

B. Efficient LCSS Distance Measure: ELCSS

Turning Function is a function which represents time series trajectories with angular distance and segment length. The angular distance is the angle made by segment with the tangent of the other segment. The angle defines a angular shape of the trajectory along with segment length. Thus, angle capture the shape of the trajectories. The angular distance between two turning function is compared and if the difference of angle between two turning function is less than or equal to threshold value, then the given segment is said to be matched. All the angular distance of turning functions are compared and similarity distance is computed. The similarity distance computed by turning function is compared with the threshold value. If similarity distance is less than or equal to threshold value, then two trajectories are said to be similar.

The two turning functions are compared using recursive function which calls itself, till terminating condition is reached. When the length of either turning function is equal to 0, then terminating condition is reached and function would returns distance. The minimum value of all possible combination is returned by the recursive function.

Given two trajectories R and S with M and N number of segments in it. R and S trajectories are converted into turning function representation and then EDWT distance measure is applied. EDWT distance measure compares the two trajectories R and S for similarity by comparing angular distance and segment length. EDWT distance measure is defined as follows:

$$ELCSS(R,S) = \min(ELCS\,S (Rest(R); Rest(S)) + match(), ELCS\,S (Rest(R), S) + g, ELCS\,S (R; Rest(S) + g)) \quad \text{if } M > 0 \text{ and } N > 0 \tag{3}$$

C. Problem Statement

Given Query trajectory $Q = (t_1, q_1), (t_2, q2),\ldots, (t_M, q_M)$, Database trajectories $D = (d_1, d_2, d3,\ldots, d_{NT})$ where $d_1, d_2, d3,\ldots, d_{NT}$ are database trajectories and a user specified threshold . The query trajectory is compared with database trajectories for similarity. The two trajectories Q and S are said to be similar if and only if similarity distance returned by ELCSS(Q,S) measure is less or equal to where Q is query trajectory and S is belong to D.

D. Properties of ELCSS measure

Our proposed distance measure supports various properties and these properties are useful for making ELCSS distance measure stronger. Following are the properties supported by ELCSS distance measure.

- Property 1: ELCSS measure always returns positive value.

$$ELCS\,S (R, S)\ 0$$

- Property 2: ELCSS measure is symmetric in nature.

$$ELCS\,S (R, S) = ELCS\,S (S, R)$$

- Property 3: ELCSS measure supports Triangular Inequality property.

$$ELCS\,S (R, P) + ELCS\,S (P, S)\ ELCS\,S (R, S)$$

- Property 4: ELCSS measure supports metric property since it supports triangular inequality, positiveness and symmetric properties.

IV. ELCSS ALGORITHM

ELCSS algorithm accepts Query trajectory Q and Database trajectories D. The query trajectory Q is compared with Database trajectories D to identify all similar trajectories as that of query. The similarity distance of query trajectory and database trajectories are computed and if the similarity distance is less or equal to threshold value, then such trajectories are returned from ELCSS algorithm.

Line 1 of ELCSS algorithm, makes a call to TurningFunc with Q as an argument. TurningFunc converts Q trajectory into turning function representation. Line 2, iterates through all the trajectories of the database. Line 3, makes a call to TurningFunc with S as an argument. Turning function converts database trajectory S into turning function representation. Line 4, makes a call to ELCSS recursive function with Q' and S' as an argument. The similarity distance is returned from ELCSS function.

Line 5, checks if similarity distance is less than or equal to threshold value, then S' is similar to query trajectory and it is stored in the ST.

Turningfunc algorithm accepts a trajectory and convert into turning function representation. Line 1, iterates through all the segments of the trajectory. Line 2, computes length of each segment. Line 3, computes angle between two consecutive segments. T' trajectory is returned from the algorithm.

ELCSS function accepts two turning functions as an argument. Line 1, check the length of the two trajectories and if length of either trajectory is zero, then return similarity distance. Line 5, makes a recursive calls with three arguments. The minimum of the three arguments is returned as the similarity distance value. Match function compares the angular distance of the current segments. If the angular distance is within the permissible limit, then zero value is returned else 1 value is returned.

ELCSS algorithm accept Query and Database trajectories as an input and it returns similar trajectories as an output. Let M and N be the number of segments in R and S trajectories respectively.

TurningFunc function converts raw trajectories into turning function representation. If there are NT number of the trajectories, then time complexity of the function is given as O(N*NT) where N is the number of segments of trajectory and NT is the number of trajectories. ELCSS function is a recursive function which can be implemented in dynamic programing with a complexity of O(N*M) where N and M are length of the trajectories. Thus, Total time complexity of ELCSS distance measure to search similar trajectories is O(N*M*NT).

1) Algorithm 1: ELCSS Function
Input: R and S : Input Trajectories
Output: Similarity Distance : SimDist

1) if Length(R) ==0 or Length(S)==0 then
2) SimDist= 1
3) return SimDist
4) else
5) SimDist= min (ELCSS(Rest(R), Rest(S))+match(),
   ELCSS (Rest(R),S) + g ,ELCSS(R,Rest(S)) +g)
6) return S imDist

2) Algorithm 2: TurningFunc Function

Input: T: Input trajectory
Output: T: Turning Function

1) for each segment S Є T do
2) T'.length=Length(S)
3) T'.angle = Angle(S)
4) return T0

3) Algorithm 3: ELCSS Algorithm

Input: Q: Query Trajectory, D: Raw Database Trajectories
Output: ST: Similar Trajectories

1) Q'=TurningFun(Q)
2) for each point S Є Di
3) S'=TurningFunc(S)
4) SimDist = ELCSS(Q',S')
5) if SimDist then
6) S is similar trajectory to Q
7) Append S to ST
8) return S T

V. EXPERIMENTAL STUDY

A. System Configuration

Experimental study was carried out on Pentium V processor with 4GB of RAM and 500GB of hard disk memory. All the programs were successfully implemented using C++ language. The g++ compiler was used to compile the C++ programs. Ubuntu 12.04 operating system was used to carry out experimental study.

B. Characteristic of Time Series Datasets

We have used real time and synthetic time series datasets for our experimental study.

1) Arrhythmia Data Set (ECG):- This database contains 279 attributes, 206 of which are linear valued and the rest are nominal. Distinguish between the presence and absence of cardiac arrhythmia and classify it in one of the 16 groups.
2) GeoLife Trajectory Dataset: This GPS trajectory dataset was collected in (Microsoft Research Asia) Geolife project contains 17,621 trajectories with a total distance of 1,251,654 kilometers and a total duration of 48,203 hours.
3) T-Drive Taxi Trajectories: A sample of trajectories from Microsoft Research T-Drive project, generated by over 10,000 taxicabs in a week of 2008 in Beijing.
4) Synthetic ECG dataset: This is a synthetic dataset contains 20,000 trajectories of 1000 patients. The program was written to simulate 1000 heart patients behavior and 20 ECGs were generated for each patient.
5) Synthetic Stock Exchange: This is a synthetic dataset contains 20,000 trajectories of 1000 companies. The program was written to simulate 1000 companies' stock exchange behavior and 20 stock trajectories were generated for each company.

C. Results and Interpretations

DTW, ERP, EDR, LCSS and ELCSS distance measures were applied to dataset 1 and their results are shown in table 4. DTW, ERP, EDR, LCSS distance measures fail to identify as proper match even though query trajectory is similar to T1, T2 and T3 trajectories. DWT, ERP distance measures similarity distance is high indicating dissimilarity between trajectories. Whereas EDR distance measure is showing high value with respect to the length of the trajectories indicating that there is mismatch of the trajectories. LCSS distance measure is showing low value indicating that there is mismatch of the trajectories. In this case, similarity distance of LCSS distance measure is zero indicating maximum dissimilarity. The results from the table reveal that ELCSS measure is able to match query trajectory Q with T1, T2 and T3 trajectories. ELCSS distance measure is able to identify similar trajectories since ELCSS distance measure is shape based measure.

Table - 3
Similarity Distance Dataset 1

Distance Measure	*(Q,T1)*	*(Q,T2)*	*(Q,T3)*
DTW	*328*	*324*	*361*
ERP	*228*	*207*	*241*
EDR	*4*	*4*	*4*
LCSS	*0*	*0*	*0*
EDTW	*0*	*0*	*0*

Table - 4
Similarity Distance Dataset 2

Distance Measure	*(Q,T1)*	*(Q,T2)*	*(Q,T3)*
DTW	*343*	*366*	*422*
EDR	*239*	*256*	*248*
ERP	*4*	*4*	*4*
LCSS	*0*	*0*	*0*
EDTW	*0*	*0*	*0*

DTW, ERP, EDR, LCSS and ELCSS distance measures were applied to dataset 2 and results are shown in table 5. This datasets was generated with noise being introduced in the dataset. DTW, ERP, EDR, LCSS distance measures fail to identify as proper match even though query trajectory is similar to T1, T2 and T3 trajectories. DWT, ERP distance measures similarity distance is high indicating dissimilarity between trajectories. Whereas EDR distance measure is showing high value with respect to the length of the trajectories indicating that there is mismatch of the trajectories. LCSS distance measure is showing low value indicating that there is mismatch of the trajectories. In this case, similarity distance of LCSS distance measure is zero indicating maximum dissimilarity. The results from the table reveals that ELCSS measure is able to match query trajectory Q with T1, T2 and T3 trajectories. ELCSS distance measure is able to match the trajectories since ELCSS distance measure is shape based and it is robust to the noise.

To compute accuracy of the distance measures, we have used 30 correct samples. Then we have applied the edit distance measures. The distance measures identify correct similar trajectories out of given 30 samples. The number of correct similar samples identified using edit distance measures were recording. Then the accuracy of edit distance measure is defined as the total number of correct samples identified divided the total number of correct samples.

Table - 5
Accuracy of Distance Measures

Distance Measure	*ECG dataset*	*Accuracy*
DTW (30)	*14*	*43*
EDR (30)	*12*	*44*
ERP (30)	*14*	*45*
LCSS (30)	*13*	*44*
EDTW (30)	*18*	*61*

Table - 6
Accuracy of Distance Measures

Distance Measure	*ECG dataset*	*Accuracy*
DTW (30)	*11*	*33*
EDR (30)	*12*	*35*
ERP (30)	*11*	*34*
LCSS (30)	*11*	*34*
EDTW (30)	*15*	*53*

Table 6 shows the accuracy of DTW, ERP, EDR, LCSS and ELCSS distance measures. The accuracy of DTW measure was 43, ERP was 44, EDR was 45 and LCSS was 44. These distance measures show low accuracy since they are not based on shape and do not support rotation invariant property. The accuracy of ELCSS distance measure was 61. The accuracy of ELCSS distance measure is at higher side since it compares trajectories based on shape feature and it is rotation invariant. Table 7 shows the accuracy of DTW, ERP, EDR, LCSS and ELCSS distance measures with noisy dataset. The accuracy of DTW measure was 33, ERP measure was 35, EDR measure 34 and LCSS was 34. These distance measures show low accuracy since they are not robust to noise. The accuracy of ELCSS distance measure was 53. The accuracy of ELCSS distance measure is at higher side since it is robust to noise.

Execution time of different edit distance measures were recorded. Execution time is the time taken to the distance measure to generate the output. Here the output is in terms of similar trajectories. Execution time of diferent distance measures was recorded for different datasets. The purpose of this experiment was to identify the performance of the edit distance measure with different datasets.

The graph in figure 4, shows the performance of the DTW, ERP and ELCSS distance measures. Since the performance of edit distance measures are almost same with respect to shape based comparison, we have carried out experimental study with DTW and ERP distance measures. Other edit distance measures can be implemented in similar line and their performance can be analyzed. The Four datasets were used to carry out experimental study such as T-drive taxi dataset, GPS trajectories, Synthetic

Stock Trajectories and Synthetic ECG Trajectories dataset. The execution time of DTW and ERP distance measures were almost same, whereas ELCSS has some extra overhead due to extra processing of turning function representation.

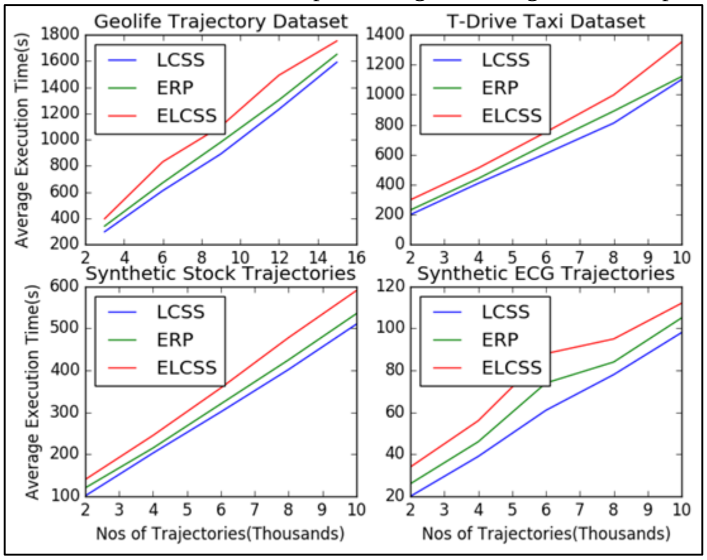

Fig. 4: Performance of Distance Measures

VI. Conclusion

We have proposed Turning Function Based Distance measure to compares time series trajectories for similarity. ELCSS distance measure compares trajectories based on the shape feature. ELCSS distance measure supports scaling, translation, rotation invariant property and robust to noise. Experimental study was carried out on real time and synthetic datasets of trajectories. Our experimental results show that ELCSS measure compares the trajectories based on shape of the trajectories and robust to the noise. The average accuracy of DTW, ERP, EDR, LCSS distance measures without noise was 44 and with noise was 34. The average accuracy of ELCSS distance measure without noise was 61 and with noise was 53. Thus, ELCSS distance measure is efficient compared to DTW, ERP, EDR, LCSS distance measures.

References

[1] R. Agrawal, C. Faloutsos, A. Swami, Ecient similarity search in sequence databases, in: FODO, Springer Verlag, 1993, pp. 69–84.
[2] R. Agrawal, K. ip Lin, H. S. Sawhney, K. Shim, Fast similarity search in the presence of noise, scaling, and translation in time-series databases, in: In-VLDB, 1995, pp. 490–501.
[3] M. H. W. Kelvin Kam Wing Chu, Fast time series searching with scaling and shifting, in: ACM-PODS, 1999, pp. 237–248.
[4] D. J. Berndt, J. Clifford, Finding patterns in time series: A dynamic programming approach, in: Advances in Knowledge Discovery and Data Mining, 1996, pp. 229–248.
[5] C. F. Yasushi Sakurai, Masatoshi Yoshikawa, Fast similarity search under the time warping distance, in: ACM-PODS, 2005, pp. 326–337.
[6] G. Das, D. Gunopulos, H. Mannila, Time-series similarity problems and well-separated geometric sets, Nord. J. Comput. 8 (4) (2001) 409–423. URL http://www.cs.helsinki.fi/njc/References/bollobas2001:409.html
[7] M. Vlachos, G. Kollios, D. Gunopulos, Discovering similar multidimensional trajectories, in: In ICDE, 2002, pp. 673–684.
[8] D. Buzan, S. Sclaroff, G. Kollios, Extraction and clustering of motion trajectories in video, in: 17th International Conference on Pattern Recognition, ICPR 2004, Cambridge, UK, August 23-26, 2004., 2004, pp. 521–524. doi:10.1109/ICPR.2004.1334287. URL http://dx.doi.org/10.1109/ICPR.2004.1334287
[9] M. Vlachos, D. Gunopulos, G. Kollios, Robust similarity measures for mobile object trajectories., in: DEXA Workshops, IEEE Computer Society, 2002, pp. 721–728.
[10] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, E. J. Keogh, Indexing multidimensional timeseries with support for multiple distance measures, VLDB J. 15 (1) (2006) 1–20.
[11] A. Dyana, S. Das, Trajectory representation using gabor features for motion-based video retrieval., Pattern Recognition Letters 30 (10) (2009) 877–892.
[12] J. J. Little, Z. Gu, Video retrieval by spatial and temporal structure of trajectories., in: M. M. Yeung, C.-S. Li, R. Lienhart (Eds.), Storage and Retrieval for Media Databases, Vol. 4315 of SPIE Proceedings, SPIE, 2001, pp. 545–552.

