Trajectory Retrieval with Latent Semantic Analysis

Apostolos N. Papadopoulos
Department of Informatics, Aristotle University
54124 Thessaloniki, GREECE
apostol@delab.csd.auth.gr

ABSTRACT

The problem of trajectory similarity has been recently attracted research interest considerably, due to its importance in diverse fields. In this work, we study trajectory similarity by attacking the problem taking an information retrieval perspective. Trajectories are first decomposed by using a grid and each trajectory is mapped to a multidimensional space where Latent Semantic Analysis is applied. Distance measures like Euclidean distance or cosine distance are applied to process similarity queries (range queries, k-NN queries). Performance evaluation results, based on real-life data sets, show the simplicity and effectiveness of the proposed scheme.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Performance

Keywords

Trajectories, LSA, query processing

1. INTRODUCTION

Trajectories appear in many application domains in diverse fields. In location-based services, a trajectory may represent the motion of a moving object (e.g., vehicle). Trajectories are archived and studied to extract useful knowledge regarding the motion behavior. In meteorological applications, a trajectory may represent the motion characteristics of a physical phenomenon (e.g., a hurricane, a storm). Investigating the motion behavior of such dangerous physical phenomena is important towards predicting future locations and therefore avoiding catastrophic results if possible. In video surveillance and tracking applications, it is important to categorize motions according to their physical meaning (e.g., a person is walking, a person is leaving a bug at the airport, an unusual behavior of a car). In the aforementioned applications, there is a definite need to archive, organize and manage the collected trajectories. The archived trajectories can be used in a number of different directions, such as: (i) trajectories are searched towards answering queries involving past locations, (ii) trajectory locations are used as a base for predicting future locations of moving objects, (iii) trajectories are organized in a convenient way to support the extraction of useful motion patterns, by means of data mining techniques (e.g., clustering, sequential pattern discovery).

With respect to the latter direction, the concept of *similarity* is of vital importance. Meaningful similarity measures result in meaningful trajectory clustering and pattern discovery. In this paper, we focus on ad-hoc similarity queries. Given a query trajectory T_q and a set of trajectories $\mathcal{T}$, the similarity query asks for the data trajectories that best match the query. A similarity query is expressed either as a range query or a k-nearest-neighbor (k-NN) query. Similarity range queries are formed by a query trajectory T_q and a user defined range r. The query asks for all trajectories that are within distance r from T_q. An alternative similarity query is the k-NN query, defined by a query trajectory T_q and an integer k, asking for the k data trajectories that best match T_q.

A trajectory is usually represented as a sequence of points in space, ordered with respect to the time instance. For example, assuming that objects move on the 2-d Euclidean plane, a trajectory T_i may be represented as follows:

$$T_i = (x_{i1}, y_{i1}, t_1), ..., (x_{in}, y_{in}, t_n), \qquad t_1 < t_2 < ... < t_n$$

where x_{ia} and y_{ia} define the location of the object at time instance t_a. The dissimilarity between two trajectories T_i and T_j is expressed by means of a distance function $D(T_i, T_j)$. It is expected that the more similar the trajectories are, the less the value $D(T_i, T_j)$ becomes and vice-versa.

In this work, we study the problem of trajectory similarity taking an information retrieval perspective. The trajectory similarity problem is transformed to that of document similarity, i.e., the trajectories are the "documents" and the space regions are the "terms". A similarity matrix is used to express inter-region proximities. This results in a weighted Euclidean distance measure, which is transformed to a non-weighted one by using specific mathematical tools. Finally, by using Latent Semantic Analysis trajectories are repre-

SAC'08 March 16-20, 2008, Fortaleza, Ceará, Brazil

sented as vectors in a low-dimensional space, where indexing schemes can be applied to speed-up retrieval. Similarity is expressed by the cosine or Euclidean distance measures in the low-dimensional space. The proposed scheme can be utilized by a geographic search engine in determining similar travel routes given a query trajectory, or in grouping travel routes in clusters according to their geographic similarity.

The rest of the work is organized as follows. Section 2 describes related research and presents the motivation behind the proposed study and our research contributions. Section 3 studies the similarity search problem from an information retrieval viewpoint. Performance results are given and discussed in Section 4, whereas Section 5 concludes the work and presents future directions.

2. RELATED WORK AND CONTRIBUTION

In many research contributions, trajectory similarity is being viewed as the multidimensional counterpart of (one-dimensional) time series similarity. One of the first studies in similarity queries for time series databases has been performed in [1]. The Discrete Fourier Transform (DFT) is used as the feature extraction method, and the Euclidean distance is used as the similarity measure.

In [7] the authors study the problem of similarity search in multidimensional data sequences, to determine similarities in image and video databases. A similarity model based on the Minkowski distance is defined, and each sequence is partitioned to subsequences by means of MBRs, to enable efficient indexing.

Another approach for expressing the similarity between two trajectories is studied in [12]. A more robust distance metric is used, based on the concept of Longest Common Subsequence (LCSS) between two trajectories. This metric is more immune to noise than the Minkowski distance. Because the proposed distance does not satisfy the metric space properties, indexing is achieved by utilizing the index structure of a hierarchical clustering algorithm.

In [14] a similarity measure between trajectories is defined, which is invariant to translation, rotation and scaling. The used measure is based on the Minkowski distance, and objects are allowed to move freely in the address space.

The aforementioned research proposals take a database perspective towards similarity query processing. However, there is important work in trajectory similarity assuming a pattern recognition perspective. In [13], a multi-object tracking system for surveillance video analysis is proposed. An efficient fuzzy clustering method is used which is based on features such as position, color and velocity. A method for clustering vehicle trajectories and discovering unusual events is studied in [6], which is based on spectral clustering. In [9], the authors study the performance of DFT-based methods for clustering trajectories that exist in a video surveillance system. They demonstrate that the use of DFT coefficients improves the clustering quality, for both k-means and SOM methods, in comparison to point-based flow vectors. Finally, [16] offers a useful performance comparison of several distance measures such as Euclidean, Dynamic Time Warping, Longest Common Subsequence, Hidden Markov Models and Hausdorff. An interesting result of this study is that in outdoor surveillance scenes the combination of Principal Component Analysis and the Euclidean distance (see [2]) offers good clustering quality.

Several research contributions are based on the use of the Euclidean distance in the original space to quantify similarities among trajectories. Although the Euclidean distance has a number of nice properties (e.g., lower bounding, easy implementation) it suffers from performance degradation due to the following reasons:

- Trajectories may have different lengths, and may have been sampled with different sample rates. This prevents the direct application of Minkowski distances and therefore re-sampling and alignment is usually applied.
- Real-life trajectories contain noise which is a direct effect of inaccuracies in tracking devices. The existence of noisy data brings up the issue of uncertainty with respect to the exact location of a moving object.
- Spatial proximity in several cases is not well preserved, although the trajectories are close in the original space.

To alleviate the aforementioned problematic phenomena, Dynamic Time Warping (DTW) has been used [15]. However, DTW-based techniques suffer for performance inefficiencies for long sequences, since the computational costs for distance calculations may increase substantially [10].

In this work, we focus on the spatial proximity of trajectories. Initially, a new grid-based distance measure is defined, the Grid Aggregate Distance (GAD), which is based on trajectory aggregation. Trajectories are mapped to a multidimensional space, whose dimensions are determined by the number of grid cells used. By using this technique we get a documents/terms analogy for trajectories. Then, by using cell similarities, the weighted Euclidean distance, and Latent Semantic Analysis, we obtain a trajectory representation in a low-dimensional space. Trajectory similarity in the transformed space is defined either by the cosine distance or the Euclidean distance.

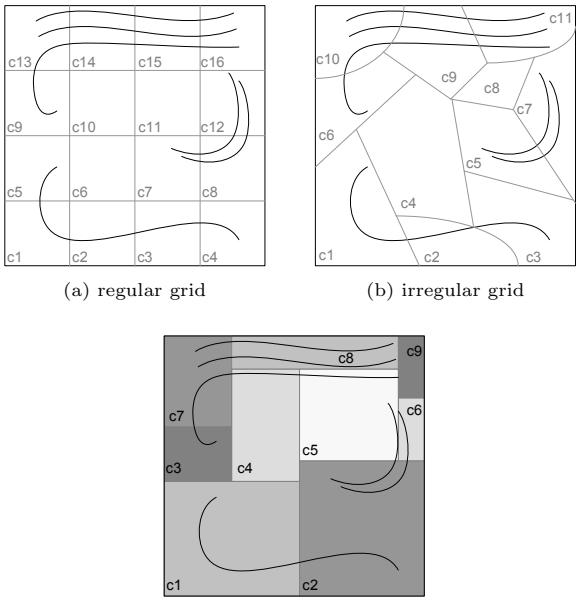
The proposed approach has a number of significant advantages: (i) it supports approximate processing, since trajectory uncertainty is compensated by the use of the grid structure, (ii) the transformed space contains a few dimensions, and therefore distance calculations require reduced computational costs, (iii) indexing is enabled by point-based or metric-based access methods, (iv) concept-based trajectory similarity is supported, something which has not been addressed so far in related research, to the best of our knowledge, and (v) existing information retrieval techniques can be applied (i.e., vector space model and LSI).

3. TRAJECTORY SIMILARITY WITH LSA

Table 1 illustrates the most important symbols used in our study, whereas the following definitions describe formally the similarity range query and the similarity k-NN query.

Definition 1 (similarity range query)
For a query trajectory T_q, a real non-negative number r and a distance measure $dist$, the range similarity query returns an answer set $\mathcal{A}$ such that:

$$\forall T_i \in \mathcal{A},\ dist(T_q, T_i) \leq r,\ \text{and}\ \forall T_j \in (\mathcal{T} - \mathcal{A}),\ dist(T_q, T_j) > r$$



(a) regular grid (b) irregular grid

(c) conceptual grid

Figure 1: Different grid types.

Definition 2 (similarity k-NN query)
For a query trajectory T_q, an integer k and a distance measure *dist*, the similarity k-NN query returns an answer set $\mathcal{A}$ of data trajectories containing at least k trajectories such that:

$$\forall T_i \in \mathcal{A}, \forall T_j \in (\mathcal{T} - \mathcal{A}), \quad dist(T_q, T_i) \leq dist(T_q, T_j)$$

The most important parameter in the above definitions is the distance measure used. Different distance measures may give totally different results. We propose to use a distance measure which is not based on the original data set, but on an aggregation performed by means of an application dependent grid structure. Figure 1 demonstrates examples of different grid types. The most common grid type is the regular (uniform) grid which is shown in Figure 1(a). Space is partitioned into cells of equal size (usually rectangular). Non-regular grids may also be applied, such as the one in Figure 1(b). For example, in an application for tracking the motion of cellular phone subscribers, one can use the existing cellular grid, where a grid cell is defined by the transmission ability of a corresponding antenna. Finally, a conceptual grid may be used, where each cell corresponds to a concept or a level of a particular measure (in contrast to the other grid types which are based on the geometric characteristics of the cells). For example, the trajectories of Figure 1(c) may correspond to a particular fish species. Assuming that gray levels represent different water temperature values, it would be interesting to express similarities among trajectories based on different temperature levels, towards determining if there are specific patterns in the data.

Symbol	Description
$\mathcal{T}$	set of data trajectories
T, T_i	data trajectories
T_q	a query trajectory
m	number of trajectories
e	number of selected eigenvalues
n	number of grid cells
k	number of requested similar trajectories
r	radius of similarity range query
c_i	the i-th grid cell
$T(c_i)$	occurrences of trajectory T in cell c_i
$GAD(T_i, T_j)$	grid aggregate distance between trajectories
$\widetilde{D}$	initial $n \times m$ data matrix of occurrences
C	the $n \times n$ cell similarity matrix
$M[i,j]$	the value of M's i-th row and j-th column

Table 1: Symbols and definitions.

3.1 The Grid Aggregate Distance

Although the proposed scheme supports all grid types discussed above, we adopt the regular grid for simplicity. However, as long as similarity among cells has been determined, any grid type can be used as well.

Let n be the number of generated grid cells. Each grid cell is labeled by an identifier c_1 through c_n. For a trajectory T, let the value $T(c_i)$ denote the number of occurrences of T in cell c_i. Note that for continuous time the value $T(c_i)$ denote the total time that T has spent in cell c_i. This way, a data matrix $\widetilde{D}$ is formed, having n rows (the number of cells) and m columns (the number of trajectories). Each trajectory T_i is transformed to the grid-based representation, and it is considered as a vector written as $\widetilde{T}_i$.

Definition 3 (data matrix)
The data matrix $\widetilde{D}$ is an $n \times m$ matrix containing occurrences of trajectories in grid cells. Each element $\widetilde{D}[i,j]$ records the number of occurrences of the j-th trajectory in the i-th cell.

The next step involves the construction of the cell similarity matrix. The similarity between two grid cells is defined by a function $sim(c_i, c_j)$ which assumes values in the interval $[0, 1]$. The similarities among all grid cells compose a matrix C (with n rows and n columns) which is symmetric and contains non-negative elements.

Definition 4 (cell similarity matrix)
The cell similarity matrix C is an $n \times n$ symmetric matrix with non-negative elements, where each matrix cell $C[i,j]$ denotes the similarity between the i-th and the j-th grid cells. More specifically, $0 \leq C[i,j] \leq 1$, and $C[i,j] = 1$ if $i = j$.

We are ready now to define the Grid Aggregate Distance between trajectories, which is expressed as a weighted Euclidean distance measure based on the data matrix $\widetilde{D}$ and the cell similarity matrix C.

Definition 5 (grid aggregate distance)
The Grid Aggregate Distance between two trajectories T_i and T_j for a cell similarity matrix C is defined as follows:

$$GAD(T_i, T_j) = (\widetilde{T}_i - \widetilde{T}_j)^T \cdot C \cdot (\widetilde{T}_i - \widetilde{T}_j) \qquad (1)$$

Since C is symmetric with real non-negative elements, it can be written as a product of three matrices as follows:

$$C = P \cdot \Delta \cdot P^T \Rightarrow C = P \cdot \Delta^{1/2} \cdot \Delta^{1/2} \cdot P^T \quad (2)$$

where Δ is a diagonal matrix, containing the eigenvalues of C, and P is a symmetric matrix for which $P^T = P^{-1}$. Furthermore, if C is positive semi-definite, its eigenvalues are non-negative. This means that the matrix $\Delta^{1/2}$ contains non-negative real values. If this is not the case, then the similarity matrix C is approximated by C_{approx} which is defined as:

$$C_{approx} = P \cdot \Delta_0 \cdot P^T \Rightarrow C_{approx} = P \cdot \Delta_0^{1/2} \cdot \Delta_0^{1/2} \cdot P^T \quad (3)$$

where Δ_0 is the diagonal matrix which contains the positive eigenvalues of C and all negative eigenvalues have been replaced by zeros. It has been proven that C_{approx} is the best approximation we can get regarding the sum of squared differences of matrices C and C_{approx} (see [11] for details). Evidently, if C is positive semi-definite then $C_{approx} = C$. By using Equations 1 and 3, the fact that $(\Delta_0^{1/2})^T = \Delta_0^{1/2}$ and elementary linear algebra manipulations, the grid aggregate distance between trajectories T_i and T_j is expressed as:

$$GAD(T_i, T_j) = (\widetilde{T}_i - \widetilde{T}_j)^T \cdot P \cdot \Delta_0^{1/2} \cdot \Delta_0^{1/2} \cdot P^T \cdot (\widetilde{T}_i - \widetilde{T}_j)$$

$$= (\widetilde{T}_i - \widetilde{T}_j)^T \cdot (\Delta_0^{1/2} \cdot P^T)^T \cdot (\Delta_0^{1/2} \cdot P^T) \cdot (\widetilde{T}_i - \widetilde{T}_j)$$

$$= \left[(\Delta_0^{1/2} \cdot P^T) \cdot (\widetilde{T}_i - \widetilde{T}_j)\right]^T \cdot \left[(\Delta_0^{1/2} \cdot P^T) \cdot (\widetilde{T}_i - \widetilde{T}_j)\right]$$

By inspecting the previous equation, it is evident that it suffices to left-multiply the data matrix $\widetilde{D}$ by $\Delta_0^{1/2} \cdot P^T$. This way, we produce the $n \times m$ matrix $\widetilde{\widetilde{D}} = \Delta_0^{1/2} \cdot P^T \cdot \widetilde{D}$ which is used to determine similarities among trajectories.

3.2 LSA and Query Processing

Up to now we have managed to express the similarity of two trajectories by means of the trajectory occurrences in each grid cell and the cell similarity matrix. Each trajectory has been mapped to a n-dimensional point, where n is the total number of grid cells. Usually, the number n is expected to be large, especially if a fine grid is used. The last step in the proposed scheme involves dimensionality reduction towards: (i) a more compact representation and (ii) the utilization of indexing schemes for fast retrieval.

LSA-based methods have been successfully applied in text information retrieval [3]. The main characteristic of this approach is that it "shifts" the documents from the term space to the concepts space. This is exactly our target, with the main difference that our data is composed of trajectories, which have been converted to the documents/terms analogy. The mathematical tool behind LSA is the Singular Value Decomposition (SVD). The matrix $\widetilde{\widetilde{D}}$ is written as a product of the matrices U, S and V as follows:

$$\widetilde{\widetilde{D}} = U \cdot S \cdot V^T$$

where the orthogonal matrix U stores the left eigenvectors, the orthogonal matrix V stores the right eigenvectors and S is a diagonal matrix containing the eigenvalues of $\widetilde{\widetilde{D}}$ in descending order. Dimensionality reduction is achieved by selecting the e largest eigenvalues (i.e., keeping the first e rows and columns of S, the first e columns of U and the first e rows of V^T). If U_e denotes the reduced matrix of left eigenvectors, each trajectory is mapped to a point in the e-dimensional space by applying the following:

$$\widetilde{\widetilde{D}}_e = U_e^T \cdot \widetilde{\widetilde{D}} \quad (4)$$

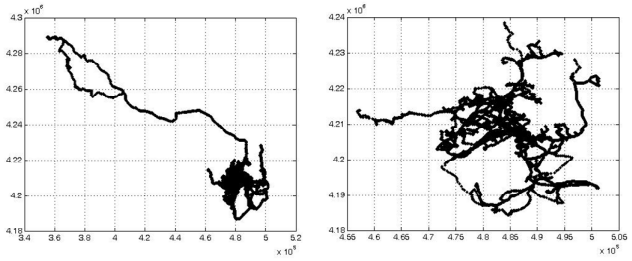
To support similarity queries efficiently, the transformed trajectories can be organized by means of spatial access methods (e.g., R-tree, X-tree) or metric access methods (e.g., M-tree, Slim-tree). When there is a new query trajectory T_q, it is first mapped to the grid representation ($\widetilde{\widetilde{T}}_q$) and then it is transformed to the e-dimensional space ($\widetilde{\widetilde{T}}_{qe}$) by using:

$$\widetilde{\widetilde{T}}_{qe} = U_e^T \cdot \widetilde{\widetilde{T}}_q \quad (5)$$

Then, the index is consulted to prune trajectories that are not possible to contribute to the final answer. According to the query type (range or k-NN) refinement should be applied towards removing the false alarms.

4. PERFORMANCE RESULTS

Two real-life data sets have been used for performance evaluation. These data sets, which are shown graphically in Figure 2, contain locations of trajectories of school buses and trucks respectively (available at http://www.rtreeportal.org). Evidently, the distribution of the data sets deviate from uniformity significantly. The BUSES data set contains 66,096 locations of school buses in the 2-dimensional space, composing 108 trajectories. The smallest trajectory contains 79 points, whereas the largest one contains 1,095 points. The TRUCKS data set contains 112,203 locations of trucks in the 2-dimensional space, composing 273 trajectories. The smallest trajectory contains 29 locations, whereas the largest one contains 992 locations.

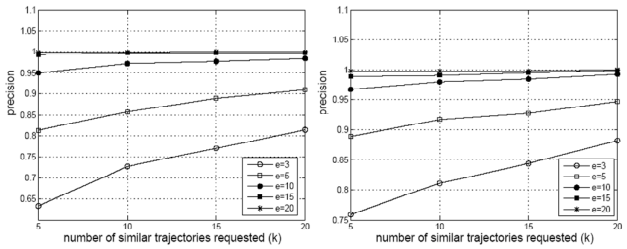


(a) BUSES data set (b) TRUCKS data set

Figure 2: Real-life trajectory data sets.

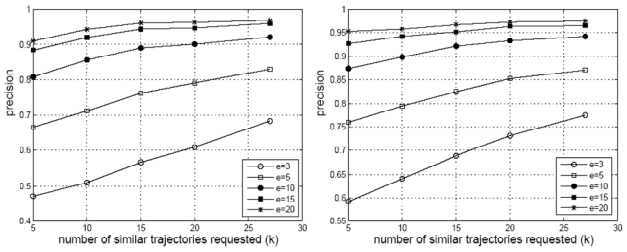
4.1 Similarity queries

Figures 3 and 4 depict the performance results for k-NN queries for BUSES and TRUCKS respectively. Note that the results have been obtained by performing the queries in the original and the transformed space without applying refinement. The graphs show the precision by varying the number k of requested trajectories and the dimensionality (number of eigenvalues) of the transformed space. Each trajectory has been set as the query, and for every query the k



(a) cosine distance (b) Euclidean distance

Figure 3: Similarity search accuracy for k-NN queries on BUSES (varying k and e).



(a) cosine distance (b) Euclidean distance

Figure 4: Similarity search accuracy for k-NN queries on TRUCKS (varying k and e).

most similar trajectories have been requested. The graphs show average precision values. A uniform grid has been used composed of 400 grid cells. Cell similarity is defined by the Euclidean distance of cell centroids. It is evident that by selecting e=10 dimensions, the recall values are very satisfactory (over 80% recall and precision) for both cosine and Euclidean distances. This is a nice result considering that no refinement has been applied to the results.

Next, we study the performance of the proposed scheme for similarity range queries. We give the results for the TRUCKS data set only. Tables 2 and 3 show the recall and precision for several values of the query radius r and the dimensionality of the transformed space. Again, 400 cells have been used whereas cell similarity is defined by the Euclidean distance of cell centroids. Note that the recall of the Euclidean distance is always 100% since the application of SVD respects the lower bounding of the Euclidean distance. On the other hand, the recall for the cosine distance may drop below 100%, denoting that some answers may be missed. By inspecting the tables it is evident that keeping e=10 or e=15 dimensions gives high recall and precision values.

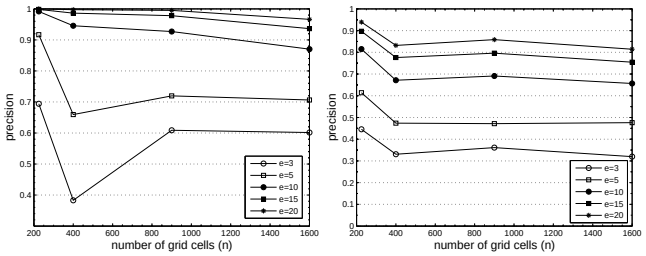
recall/precision			
radius(r)	e=5	e=10	e=15
0.1	1.0/0.4185	1.0/0.6384	1.0/0.7661
0.2	1.0/0.5509	1.0/0.7216	1.0/0.8197
0.5	0.9961/0.8164	1.0/0.9144	1.0/0.9469
1.0	0.9853/0.9799	0.9962/0.9954	0.9980/0.9975
1.5	0.9724/0.9999	0.9895/1.0	0.9938/1.0

Table 2: Results for range queries on TRUCKS (cosine distance, max distance is 1.9).

recall/precision			
radius(r)	e=5	e=10	e=15
10	1.0/0.9023	1.0/0.9860	1.0/0.9890
20	1.0/0.5882	1.0/0.8483	1.0/0.9408
50	1.0/0.5820	1.0/0.7679	1.0/0.8673
100	1.0/0.8609	1.0/0.9414	1.0/0.9654
200	1.0/0.9876	1.0/0.9962	1.0/0.9986

Table 3: Results for range queries on TRUCKS (Euclidean distance, max distance is 683).

Finally, we present some representative results showing the impact of the number of grid cells used for the grid formulation. The precision results for the Euclidean distance (recall is always 100%) are given in Figure 5. The results correspond to similarity range queries. The radius of the query in each case has been selected so as to retrieve approximately 10 trajectories in the answer set. Evidently, the more dimensions we use, the better the precision becomes. However, by using e=10 or e=15 dimensions in the transformed space, adequate precision is achieved, which means that the number of false alarms is very low.



(a) BUSES (b) TRUCKS

Figure 5: Precision by varying the number of cells and the number of eigenvalues.

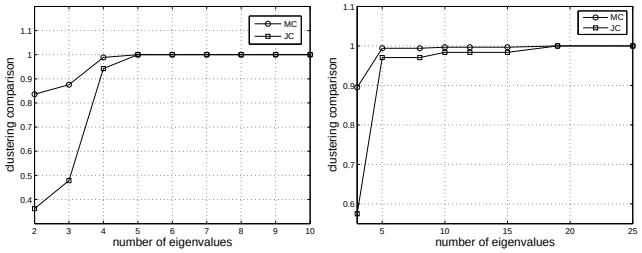
4.2 Clustering Comparison

In this section, we investigate the effectiveness of the LSA scheme on clustering (without loss of generality, K-means has been used as the clustering algorithm). This is achieved by comparing the clustering performed prior to SVD to the clustering performed after dimensionality reduction. The clustering effectiveness is measured by means of the Matching (MC) and the Jaccard Coefficient (JC) of the clusterings [5]:

$$MC = \frac{N_{00} + N_{11}}{N_{00} + N_{11} + N_{01} + N_{10}} \quad \text{and} \quad JC = \frac{N_{11}}{N_{11} + N_{01} + N_{10}}$$

where N_{00} is the number of trajectory pairs that are in different clusters both in both clusterings, N_{11} is the number of trajectory pairs that are in the same cluster in both clusterings, N_{01} is the number of trajectory pairs that are in different clusters in the first clustering but are in the same cluster in the second, and N_{10} is the number of trajectory pairs that are in the same cluster in the first clustering but in different clusters in the second.

Figure 6 illustrates the clustering comparison results for the TRUCKS data set, by using both cosine (a) and Eu-



(a) cosine dist. (6 clusters) (b) Euclidean dist. (8 clusters)

Figure 6: Clustering comparison for TRUCKS.

clidean distance (b). The number of clusters for each distance measure has been selected in order to give a relatively high silhouette coefficient. It is evident that both MC and JC assume values larger than 0.9 when five or more eigenvalues are used for the trajectory representation. Therefore, instead of performing a clustering in the n-dimensional space (where n is the number of grid cells) we can apply clustering on the e-dimensional space (where e is the number of eigenvalues maintained). Evidently, the number of retained eignevalues depends on the characteristics of each data set.

5. CONCLUSIONS

Trajectories appear in several applications domains such as location-based services, monitoring of meteorological phenomena, logistics, video tracking/surveillance, animal tracking. In this work, we have studied the trajectory similarity problem from an information retrieval perspective. Trajectories are first mapped to a grid representation, where LSA is applied towards "shifting" trajectories to the concepts space. The proposed scheme is capable of handling spatial proximity, uncertainty and it allows concept-based trajectory similarity. The performance results have shown that not only similarity queries can be efficiently handled, but also clustering can be performed in the transformed space since distances are well preserved. Some directions for future research are: (i) the study of alternative dimensionality reduction methods, (ii) the performance comparison of different indexing schemes, and (iii) the combination of geographic with textual information to define similarity among trajectories. Regarding the last issue, each route may be annotated with textual information. Combining the two types of information (geographic and textual) is an interesting and very promising research topic [4, 8].

6. REFERENCES

[1] R. Agrawal, C. Faloutsos, A. Swami: "Efficient Similarity Search in Sequence Databases", *Proceeedings of the International Conference on Foundations of Data Organization (FODO)*, pp.69-84, 1993.

[2] F.I. Bashir, A.A. Khokhar, D. Schonfeld: "Segmented Trajectory Based Indexing and Retrieval of Video Data", *Proceedings of the Iternational Conference on Image Processing (ICIP)*, Vol.2, pp.623-626, 2003.

[3] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landaurer, R. Harsjman: "Indexing by Latent Semantic Analysis", *Journal of the American Society for Information Sciences*, Vol.41, pp.301-407, 1990.

[4] A.R. Doherty, C. Gurrin, G.J.F. Jones, A.F. Smeaton: "Retrieval of Similar Travel Routes using GPS Tracklog Place Names", *Proceedings of the 3rd Workshop on Geographic Information Retrieval (GIR)*, 2006.

[5] H. Finch: "Comparison of Distance Measures in Cluster Analysis with Dichotomous Data", *Journal of Data Sciences*, Vol.3, pp.85-100, 2005.

[6] Z. Fu, W. Hu, T. Tan: "Similarity Based Vehicle Trajectory Clustering and Anomaly Detection", *Proceedings of the International Conference in Image Processing (ICIP)*, Vol.2, pp.602-605, 2005.

[7] S.-L.Lee, S.-J. Chun, D.-H. Kim, J.-H. Lee, C.-W. Chung: "Similarity Search for Multidimensional Data Sequences", *Proceedings of the 16th International Conference on Data Engineering (ICDE)*, pp.599-608, 2000.

[8] B. Martins, M.J. Silva, S. Freitas, A.P. Afonso: "Handling Locations in Search Engine Queries", *Proceedings of the 3rd Workshop on Geographic Information Retrieval (GIR)*, 2006.

[9] A. Naftel, S. Khalid: "Motion Trajectory Learning in the DFT-Coefficient Feature Space", *Proceedings of the 4th IEEE International Conference on Computer Vision Systems (ICVS)*, 2006.

[10] S. Park, W.W. Chu, J. Yoon, C. Hsu: "Efficient Searches for Similar Subsequences of Different Lengths in Sequence Databases", *Proceedings of the 16-th International Conference on Data Engineering (ICDE)*, pp.23-32, 2000.

[11] J. Ponce, "On Computing Metric Upgrades of Projective Reconstructions Under the Rectangular Pixel Assumption", *Revised papers from the 2nd European Workshop on 3D Structure from Multiple Images of Large-Scale Environments*, pp.52-67, 2000.

[12] M. Vlachos, G. Kollios, D. Gunopulos: "Discovering Similar Multidimensional Trajectories", *Proceedings of the 18th IEEE International Conference on Data Engineering (ICDE)*, pp.673-684, 2002.

[13] D. Xie, W. Hu, T. Tan, J. Peng: "A Multi-Object Tracking System for Surveillance Video Analysis", *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, Vol.4, pp.767-770, 2004.

[14] Y. Yanagisawa, J.-I. Akahani, T. Satoh: "Shape-Based Similarity Query for Trajectory of Mobile Objects", *Proceedings of the 4th International Conference on Mobile Data Management (MDM)*, pp.63-77, 2003.

[15] B.-K. Yi, H. V. Jagadish, C. Faloutsos: "Efficient Retrieval of Similar Time Sequences Under Time Warping", *Proceedings of the 14th International Conference on Data Engineering (ICDE)*, pp.201-208, 1998.

[16] Z. Zhang, K. Huang, T. Tan: "Comparison of Similarity Measures for Trajectory Clustering in Outdoor Surveillance Scenes", *Proceedings of the 18th International Conference on Pattern Recognition (ICPR)*, Vol.3, pp.1135-1138, 2006.