Low Complexity Spatial Similarity Measure of GPS Trajectories

Radu Mariescu-Istodor, Andrei Tabarcea, Rahim Saeidi and Pasi Fränti

Speech and Image Processing Unit, School of Computing, University of Eastern Finland, Joensuu, Finland
{radum, tabarcea, rahim, franti}@cs.uef.fi

Keywords: GPS Trajectory, Spatial Similarity, MGRS, Cell Approximation, Sampling Frequency, Interpolation, Dilation.

Abstract: We attack the problem of trajectory similarity by approximating the trajectories using a geographical grid based on the MGRS 2D coordinate system. We propose a spatial similarity measure which is computationally feasible for big data collections. The proposed measure is based on cell matching with a similarity metric drawn from Jaccard index. We equip the proposed method with interpolation and dilation to overcome the problems missing data and different sampling frequencies when comparing two trajectories. The proposed measure is implemented online in the framework of Mopsi[a].

[a] cs.uef.fi/mopsi

1 INTRODUCTION

In recent years, GPS technology has been widely available in consumer devices, especially in smartphones[1], which count as more than a half on total mobile phone sales[2]. Furthermore, most of the users utilize their phone to find their location, amongst other services[3]. The wide availability of GPS-enabled smartphones that are also connected to the Internet has made the collection of large amount of location-based data possible. Such data includes geo-tagged photos, videos and geographical trajectories. Collecting geographical trajectories has practical applications in fleet management, sports tracking, recommending tourist trajectories, improving navigation and determining mobility patterns.

Having a large-scale collection of GPS trajectories raises the challenge of how to organize the data, how to present it in a meaningful way and how to filter out irrelevant data. Computing *trajectory similarity* is a tool that can be used in addressing those challenges (Agrawal et al., 1993). A problem in computing similarity of GPS trajectories is that the large amount of data does not permit processing raw trajectories in real time.

Time series analysis of one-dimensional data across the time has been used for analyzing stock changes, weather data and biomedical measurements (Hamilton, 1994; Chan and Fu, 1999; Worsley and Friston, 1995; Lange and Naumann, 2011). Despite the significant research output on time series analysis, the concept of computing similarity for traces of moving objects in the framework of spatio-temporal databases has been studied much less. Finding *k-nearest* trajectories, indexing and clustering of spatio-temporal data are among the recent directions of research with many applications to make queries in moving object databases (Frentzos et al., 2007a; Ni and Ravishankar, 2007; Frentzos et al., 2007b; Güting et al., 2010; Pelekis et al., 2011). These algorithms can be applied also for measuring the trajectory similarity (Hu and Steenkiste, 2006).

Using *Euclidean distance* is not practical for the case that the length of two trajectories are not equal (Yanagisawa et al., 2003). *Dynamic time warping* handles matching two sequences of different length but it is very sensitive to noisy data (Berndt and Clifford, 1994). Algorithms like *longest common subsequence* (LCS) (Vlachos et al., 2002b; Vlachos et al., 2002a) or *edit distance on real sequence* (EDR) (Chen et al., 2005) are designed to account for noisy and missing data but they are not perturbation free. Considering M trajectories of N points on average, the computational complexity of these algorithms is at minimum $O(M^2 \cdot N^2)$. Hence, these algorithms cannot provide real-time results when dealing with a large collection of data.

These algorithms do not utilize time stamps. By

[1] abiresearch.com/research/product/1005746-mobile-device-user-interfaces

[2] gartner.com/newsroom/id/2623415

[3] pewinternet.org/Reports/2012/Location-based-servi ces.aspx

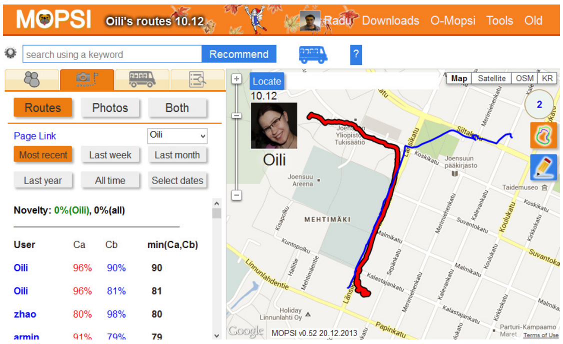
using the timing information a complete movement profile can be provided and the similarity of two trajectories can be used in trajectory clustering applications. The similarity measurement in LCS and EDR are based on point-to-point distance calculations. In the event of having two trajectories with different sampling frequency, LCS and EDR cannot provide correct similarity measure (Frentzos et al., 2007b). Although it is always possible to use a trajectory reduction or approximation algorithm to represent a trajectory with far less representatives for similarity calculation, the quality of such an approximation algorithm and overhead computational complexity is debatable (Ni and Ravishankar, 2007).

In this paper, we propose a fast method of computing trajectory similarity by approximating the trajectories using a geographical grid based on a 2D coordinate system. This process reduces a trajectory from points to cells with order of magnitude less details in representation and subsequently in distance calculations. We employ an asymmetric similarity metric inspired by Jaccard index. Dealing with GPS data collection, it is common to have bunch of data points lost or compare trajectories traveled by car with walking speed trajectories. We propose interpolation and dilation of trajectories represented as cells to overcome these difficulties. In the results section we simulate missing data and trajectory sampling frequency mismatch with two example trajectories and demonstrate the efficiency of the proposed approach. Conclusions are drawn after the discussion of results.

2 MOPSI

Mopsi is a research project location-based service developed at the University of Eastern Finland by Speech and Image Processing Group from the School of Computing. (Fränti et al., 2011) Mopsi offers multiple applications of location-aware systems, being a test-bed for various research topics that involve location-aware data. It contains tools for collecting, processing and displaying location-based data, such as photos or trajectories, along with social media integration. The main topics addressed in Mopsi are collecting location-based data, mining location data from web pages, processing, storing and compressing of GPS trajectories, detecting transportation mode from GPS trajectories, recommending *points of interest*, using location information in social networks, detecting users actions by using their location and building location-based games with the help of user-generated collections.

Location-based data is very common among web-pages, especially when their content describe commercial services, landmarks or public institutions. However, the location data is more commonly presented in a human-readable way and not as geographical coordinates, which are more accurate and easier to be automatically identified. We propose a method to automatically identify location information from web-pages by detecting postal addresses (Fränti et al., 2010).



Novelty: 0%(Oili), 0%(all)

User	Ca	Cb	min(Ca,Cb)
Oili	96%	90%	90
Oili	96%	81%	81
zhao	80%	98%	80
armin	91%	79%	79

Figure 1: Mopsi application on web showing an example of two trajectories which display a common region.

Mopsi provides tools to collect GPS trajectories and it includes more than 9000 trajectories composed of over 7 million points by the end of 2013. Mopsi uses fast retrieval and displaying of the data (Waga et al., 2013) based on GPS trajectory polygonal approximation (Chen et al., 2012a). GPS trajectories are also compressed for optimizing storage space (Chen et al., 2012b). Transport mode information can be also retrieved by automatically analyzing GPS trajectories (Waga et al., 2012). The algorithm uses a second order Markov model to segment the trajectories and to detect car, bicycle, running or walking transportation modes.

The relevance of location-based media can be assessed by considering several aspects such as time, location, content or social network (Fränti et al., 2011), which are used to create a context for each user. A personalized recommender system can recommend relevant data based on user location and user context (Waga et al., 2011). Such data can be geotagged photos, services confirmed by administrators or GPS trajectories. Users can share their location in real-time by using mobile phone location-aware applications. This allows for the detection of various location-based actions such as meetings, visiting or passing-by *points of interest* (Mariescu-Istodor, 2013). Mopsi also includes location-based games, such as O-Mopsi (Tabarcea et al., 2013), which is an orienteering game using the data from a user-generated photo collection.

Mopsi provides tools for collecting location-based

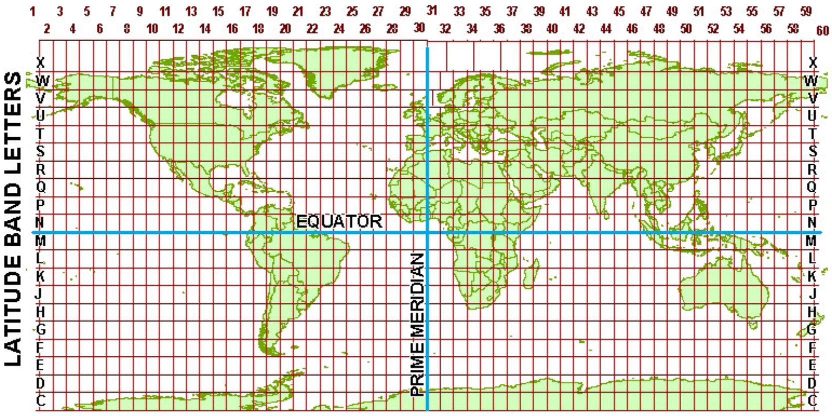


Figure 2: MGRS grid zones (source[4]).

data with mobile devices. It is available on most mobile operating systems (Android, iOS, Windows Phone, Symbian). The server-side processes the data collected by the user and displays the data collection. It also provides social features and integration which social media, with functionalities such as chatting, friends tracking and sharing data to Facebook. The Mopsi routes module provides tools for trajectory recording and displaying the large amount of data in reasonable time. Trajectory similarity is the newest addition to the Mopsi routes module.

3 TRAJECTORIES

In Mopsi we record a user's location at a certain time as a point $\mathbf{p}_k = (x_k, y_k, t_k)$, where x_k is the latitude, y_k is the longitude and t_k is the timestamp of point k. An ordered sequence of these points, defines a spatial trajectory $\mathbf{R} = (\mathbf{p}_1, \ldots, \mathbf{p}_K)$. We calculate the similarity between a reference trajectory $\mathbf{R}_a$ and all the other $M-1$ trajectories in the database, $\mathbf{R}_m$, $m = 1, \ldots, M$.

The similarity of two trajectories can be calculated as the Jaccard index:

$$J(\mathbf{R}_a, \mathbf{R}_m) = \frac{|\mathbf{R}_a \cap \mathbf{R}_m|}{|\mathbf{R}_a \cup \mathbf{R}_m|}, \tag{1}$$

Instead of this symmetric measure we want to find out if the reference trajectory is completely covered by another trajectory. Thus, we consider the following asymmetric similarity metric:

$$Sim(\mathbf{R}_a, \mathbf{R}_m) = \frac{|\mathbf{R}_a \cap \mathbf{R}_m|}{|\mathbf{R}_a|}, \tag{2}$$

$$Sim(\mathbf{R}_m, \mathbf{R}_a) = \frac{|\mathbf{R}_a \cap \mathbf{R}_m|}{|\mathbf{R}_m|}. \tag{3}$$

The first one shows what percentage of $\mathbf{R}_a$ is shared with $\mathbf{R}_m$ and the second shows what percentage of $\mathbf{R}_m$ is shared by $\mathbf{R}_a$. The way that we perform intersection operator is described in the following sections after we quantize the trajectories into cells.

3.1 Cell Approximation

In a preprocessing step, we generate a cell representation for a trajectory after it has been recorded. The Military Grid Reference Systems (MGRS) is an alpha-numeric system for expressing UTM/UPS coordinates. MGRS is used by NATO to locate points on earth. A single alpha-numeric value references a position that is unique for the entire earth (see Figure 2). MGRS is a projected coordinate system which uses a 2-dimensional Cartesian horizontal position orientation, so that locations are identified independently of vertical position. MGRS shares several characteristics with UTM such as the division of earth into projection zones and using easting and northing in meters within a designated zone. The main differences are that a MGRS zone is a 100km square within a UTM zone, whilst a UTM zone is usually 6 degrees in east-west and 8 degrees in north-south area and also that the notation of the areas is different. Based on the coordinate resolution, MGRS can define a grid with square cells with the length starting from 100km up to 10m or even 1m.

We approximate a trajectory $\mathbf{R} = \{(x_k, y_k)\}_{k=1}^{K}$ by a sparse binary matrix representation $\mathbf{C}$ where,

$$(\mathbf{C})_{ij} = \begin{cases} 1 & 0 < x_k - iL < L, 0 < y_k - jL < L \\ 0 & \text{Otherwise} \end{cases}, \tag{4}$$

where L stands for the cell length (25 meters in this paper) and indexes i and j span over in horizontal and vertical cells that trajectory R is residing inside. Figure 3 shows how the reference trajectory is approxi-

[4] earth-info.nga.mil/GandG/coordsys/grids/universal_grid_system.html

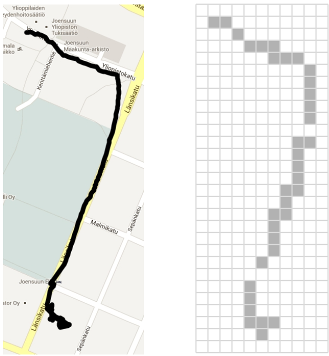


Figure 3: Example of a trajectory of 420 points being represented by 35 cells using the approximation in Equation 4. The cell representation is not continuous. The gaps appear because of the fixed cell size, variations in movement speed (or different sampling frequencies) and missing GPS locations. It is likely for such gaps to appear especially when users are moving by car, train or plane.

mated by cells. Generating the cell representation for a trajectory of average length of N points is done in $O(N)$ time.

3.2 Measuring Similarity

The similarity between two trajectories $\mathbf{R}_a, \mathbf{R}_m$ can now be calculated as:

$$Sim(\mathbf{R}_a, \mathbf{R}_m) = \frac{\|\mathbf{C}_a \odot \mathbf{C}_m\|_0}{\|\mathbf{C}_a\|_0}, \quad (5)$$

where $\mathbf{C}_a$ and $\mathbf{C}_m$ are the cell representations of $\mathbf{R}_a$ and $\mathbf{R}_m$, respectively, $\mathbf{C}_a \odot \mathbf{C}_m$ is a *Hadamard product* of two matrices $\mathbf{C}_a$ and $\mathbf{C}_m$ defined as $(\mathbf{C}_a \odot \mathbf{C}_m)_{ij} = (\mathbf{C}_a)_{ij} \cdot (\mathbf{C}_m)_{ij}$ and $\|\mathbf{C}\|_0$ represents the ℓ_0-quasinorm. In implementation, $\mathbf{C}_a$ and $\mathbf{C}_m$ are multiplied element by element and then we measure the number of non-zero elements. Figure 4 shows two sample trajectories being matched.

Assuming we have the cell representation $\mathbf{C}$ of a reference trajectory $\mathbf{R}$ we calculate the similarity for all trajectories in database in two steps. First, we find all the trajectories which share at least one cell with the reference trajectory. This has a time complexity of $O(N' \cdot (q + M'))$ where q represents the steps needed by the database system to perform the search ($N' \ll N$ and $M' \ll M$). In contrast to the average length N of a trajectory $\mathbf{R}$, we define $N' = \|\mathbf{C}\|_0$ as the number of non-zero elements in cell-approximated version of $\mathbf{R}$. In a similar way, M' indicates the number of other trajectories that share at least one cell with trajectory $\mathbf{R}$. Secondly, we calculate the trajectory similarity according to Equation (5) with a time complexity of $O(M' \cdot N')$. The overall complexity of the similarity scoring is $O(M' \cdot N')$ (assuming q constant by adding a proper indexing structure in the database).

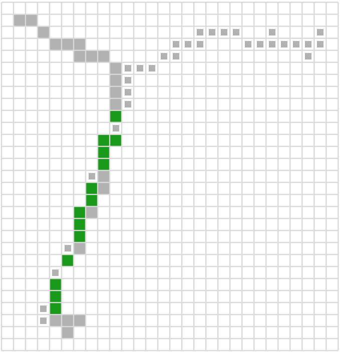


Figure 4: Matching two trajectories using the cell representation. The green cells denote the reference trajectory and the gray cells represent the other trajectory. The ’x’ symbol is used to mark the cells shared by two trajectories; $Sim(\mathbf{C}_a, \mathbf{C}_m) = 40\%$ and $Sim(\mathbf{C}_m, \mathbf{C}_a) = 31\%$
.

In Figure 4 the straighforward application of the similarity scores yield similarity scores of 40% and 31% even though the trajectories seem to have more than 50% similarity by visual inspection. In the next subsections we analyze why this happens.

3.3 Interpolation

When the user is traveling fast or when recording frequency is low we notice gaps in the trajectory representation by cells. Gaps can also appear due to lack of GPS signal. Figure 5 shows three examples when different sized gaps appear in the cell representation of a trajectory. In cell approximation stage in Equation 4, we process the trajectory data points in the sequence they are recorded. In this way, the sequence of cells being detected as “1s” are used to determine if the next cell is connected to the current cell and find a potential gap in cell-approximation.

In order to fill the gap, the line equation between two cells is obtained from the start and end points as

$$j = f(i) = \frac{j_2 - j_1}{i_2 - i_1}(i - i_1) + j_1 \quad (6)$$

where i_1 and j_1 are the coordinates of one cell and i_2, j_2 are the coordinates of the other cell. The line in Equation 6 is then sampled by the cells that it is passing through and then set respective cell values as $(\mathbf{C}_{ij}) = 1$.

By performing interpolation, the trajectory similarity presented in Figure 4 is now updated as plotted in Figure 6. The similarity values are still below the visual expectations. The reason is that two cell representations may not overlap even though the trajectories are close to each other.

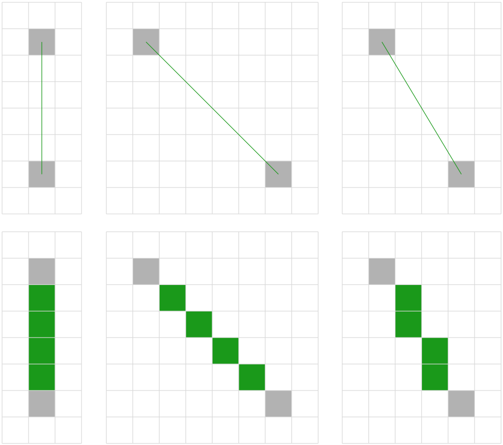


Figure 5: Interpolation between two cells in order to fill a gap; three example situations are depicted.

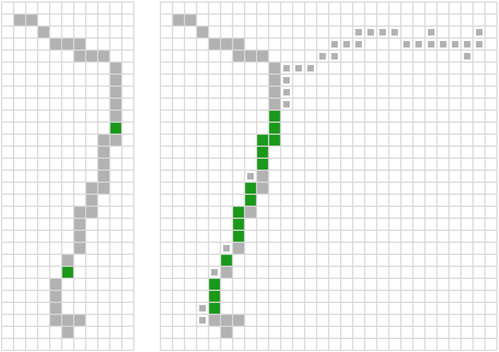


Figure 6: The trajectory having gaps is interpolated and the matching of the two trajectories becomes: $Sim(\mathbf{C}_a, \mathbf{C}_m) = 41\%$ and $Sim(\mathbf{C}_m, \mathbf{C}_a) = 33\%$.

3.4 Dilation

A frequent situation is that two nearby trajectory segments are evolving along each other in cell representation instead of overlapping. An example is provided in Figure 7 We solve this issue by applying *morphological dilation* on the trajectories and taking into account the neighbouring cells of a trajectory. We define $\mathbf{C}^d$ as a result of *binary dilation* of sparse binary representation $\mathbf{C}$ by *binary structure* $\mathbf{S}$ with

$$\mathbf{C}^d = \mathbf{C} \oplus \mathbf{S} = T(\mathbf{C} * \mathbf{S}), \tag{7}$$

where $\oplus$ defines the binary dilation and $*$ indicates the convolution operator. In the Equation 7, $T(\cdot)$ stands for *binarization transform* as

$$T((\mathbf{C} * \mathbf{S})_{ij}) = \begin{cases} 0 & 0 \leq (\mathbf{C} * \mathbf{S})_{ij} < 1 \\ 1 & \text{Otherwise} \end{cases} \tag{8}$$

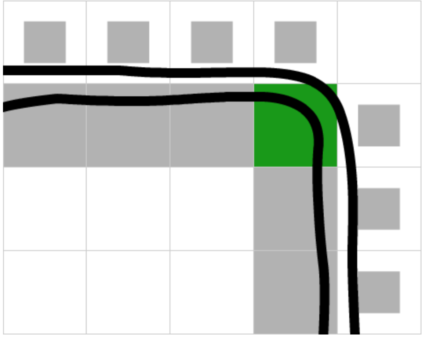


Figure 7: We see that two trajectories which are close enough to be considered similar can be represented by different cells. Only a single cell is shared by the cell representation of the two trajectories.

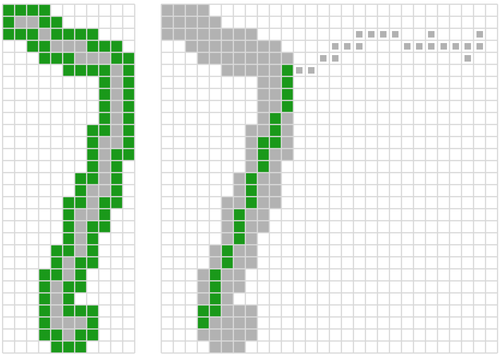


Figure 8: The reference trajectory is dilated and the matching of the two trajectories becomes: $Sim(\mathbf{C}_a, \mathbf{C}_m) = 64\%$ and $Sim(\mathbf{C}_m, \mathbf{C}_a) = 53\%$.

Figure 8 shows how a trajectory is dilated with the following structure

$$\mathbf{S} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \tag{9}$$

Then the two trajectories are matched when one of the trajectories is dilated. The similarity score is now calculated with $\mathbf{C}_a$ and $\mathbf{C}_m^d$ as in Equation 5. Typically the number of cells used in the trajectory representations increases by a factor of 3 when dilation is applied.

4 RESULTS

We implement our method in a real-world application, as a prototype using the Mopsi project route analysis module [5]. We investigate issues that may appear

[5] cs.uef.fi/mopsi/?tab=routes&userId=13&routeId=1378824019381&similarity=true

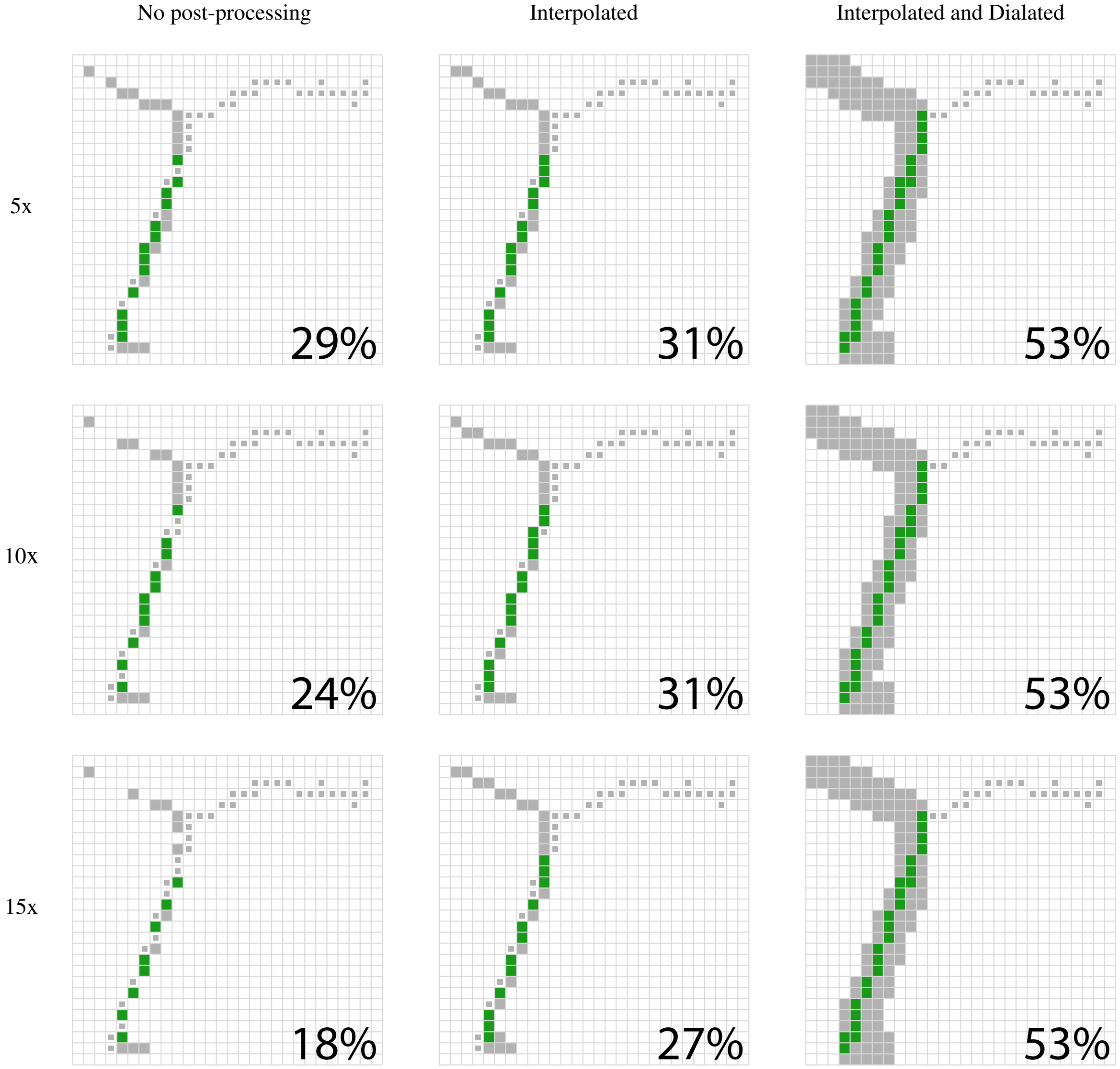


Figure 9: Simulating different sampling frequencies by subsampling the reference trajectory with a factor of 5x, 10x and 15x.

when collecting GPS trajectories in a practical application such as different sample rates, interpolation of collected points or breaks in the GPS signal caused by technical or environmental problems.

Firstly, as shown in Figure 9, we investigate how a different sampling frequency impacts the similarity score calculation. The reference trajectory is subsampled with factor f by only keeping every f^{th} element from the original trajectory. We notice that the interpolation step doesn't increase the similarity scores significantly. However, when followed by dilation, the similarity score indicates robustness against variations in sampling frequency which is a desired property for a trajectory matching procedure.

The other common issue while recording a trajectory is loss of location information for a brief period of time. This can happen, for example, if the user goes through a building, a tunnel or simply due to device software error. We simulate this behavior and see how the similarity scoring is affected in Figure 10. When removing 90 points we notice that the similarity score has dropped even when using interpolation and dilation. This happened because we removed a significant amount of subsequent points (20% of the trajectory). Interpolation does not have enough information to reconstruct the trajectory appropriately and consequently, loss of many data points in a trajectory is detrimental for similarity calculations.

The proposed method is implemented in two steps for real-world application: the preprocessing step,

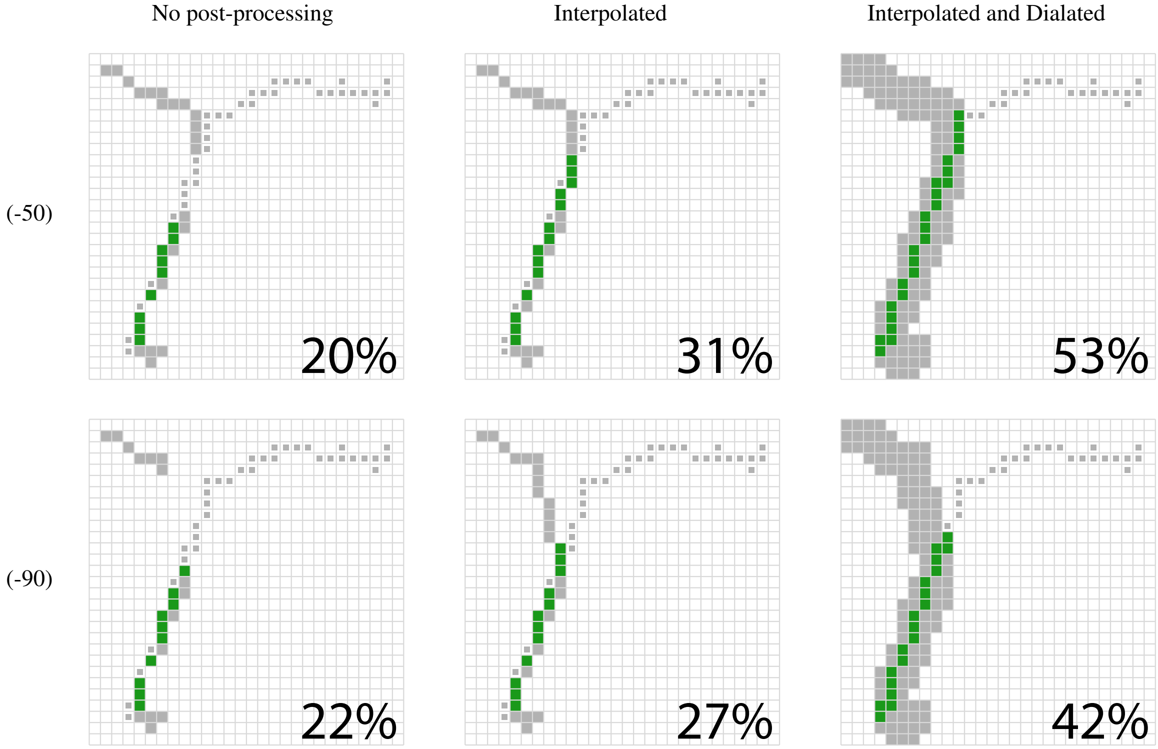


Figure 10: Simulating loss of GPS signal by removing 50 and 90 sequential points from the reference trajectory.

done when a new trajectory is added into the system and the similarity score calculation step, performed when searching all the similar trajectories of a given trajectory. When not using interpolation or dilation the time complexity for the preprocessing step is $O(M \cdot N)$ for M trajectories of average length N points. The similarity score calculation has a time complexity of $O(M' \cdot N')$. After interpolation is applied there will be an increase on the N' and M' parameters which increase, however, stay at the same order of magnitude. N' increases by the number of cells added trough interpolation and M' increases by the number of trajectories that share at least one cell with interpolated trajectory. The dilation stage increases the N' and M' parameters once more. N' typically increases by a factor of 3 and M' grows by the number of trajectories that share the cells that are added to the representation as a result of dilation. The overall complexity for M trajectories in the database is governed by $O(M \cdot N)$ for cell approximation and $O(\alpha \cdot M \cdot M' \cdot N')$ for similarity score calculation including interpolation and dilation ($\alpha \approx 6$, $M' \ll M$, $N' \ll N$). The similarity cell approximation complexity of $O(M \cdot N)$ is negligible compared to $O(\alpha \cdot M \cdot M' \cdot N')$ for score calculation. Hence, the overall computational complexity of the proposed approach is dominated by $O(\alpha \cdot M \cdot M' \cdot N')$ which is comparably much less than $O(M^2 \cdot N^2)$ for other similarity metrics presented in section 1.

5 CONCLUSIONS

We presented a method for computing similarity between trajectories in a large data collection. Because trajectories are likely to have different speed profile and missing points, interpolation and dilation techniques are employed before the scoring. We have demonstrated that the method is robust except when many points are removed and dramatically affect the structure of a trajectory. In that situation there is simply not enough information to rebuild the path and provide correct similarity values. The method was implemented in Mopsi, where for a given trajectory we display a list of similar paths in reverse order of the similarity scores.

REFERENCES

Agrawal, R., Faloutsos, C., and Swami, A. (1993). *Efficient similarity search in sequence databases*. Springer.

Berndt, D. J. and Clifford, J. (1994). Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370. Seattle, WA.

Chan, K.-P. and Fu, A. W.-C. (1999). Efficient time series matching by wavelets. In *Data Engineering, 1999. Proceedings., 15th International Conference on*, pages 126–133. IEEE.

Chen, L., Özsu, M. T., and Oria, V. (2005). Robust and fast similarity search for moving object trajectories. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 491–502. ACM.

Chen, M., Xu, M., and Fränti, P. (2012a). Compression of gps trajectories. In *Data Compression Conference (DCC), 2012*, pages 62–71. IEEE.

Chen, M., Xu, M., and Fränti, P. (2012b). A fast $O(N)$ multiresolution polygonal approximation algorithm for GPS trajectory simplification. *IEEE Transactions on Image Processing*, pages 2770–2785.

Fränti, P., Chen, J., and Tabarcea, A. (2011). Four aspects of relevance in location-based media: content, time, location and network. In *Web Information Systems and Technologies (WEBIST'11), International Conference on*, pages 413–417.

Fränti, P., Tabarcea, A., Kuittinen, J., and Hautamäki, V. (2010). Location-based search engine for multimedia phones. In *Multimedia and Expo (ICME), 2010 IEEE International Conference on*, pages 558–563. IEEE.

Frentzos, E., Gratsias, K., Pelekis, N., and Theodoridis, Y. (2007a). Algorithms for nearest neighbor search on moving object trajectories. *Geoinformatica*, 11(2):159–193.

Frentzos, E., Gratsias, K., and Theodoridis, Y. (2007b). Index-based most similar trajectory search. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pages 816–825. IEEE.

Güting, R. H., Behr, T., and Xu, J. (2010). Efficient k-nearest neighbor search on moving object trajectories. *The VLDB Journal*, 19(5):687–714.

Hamilton, J. D. (1994). *Time series analysis*, volume 2. Cambridge Univ Press.

Hu, N. and Steenkiste, P. (2006). Quantifying internet end-to-end route similarity. In *Passive and Active Measurement Conference*, volume 2006, pages 101–110.

Lange, D. and Naumann, F. (2011). Efficient similarity search: arbitrary similarity measures, arbitrary composition. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1679–1688. ACM.

Mariescu-Istodor, R. (2013). Detecting user actions in MOPSI. Master's thesis, University of Eastern Finland.

Ni, J. and Ravishankar, C. V. (2007). Indexing spatio-temporal trajectories with efficient polynomial approximations. *Knowledge and Data Engineering, IEEE Transactions on*, 19(5):663–678.

Pelekis, N., Kopanakis, I., Kotsifakos, E. E., Frentzos, E., and Theodoridis, Y. (2011). Clustering uncertain trajectories. *Knowledge and Information Systems*, 28(1):117–147.

Tabarcea, A., Wan, Z., Waga, K., and Fränti, P. (2013). O-mopsi: Mobile orienteering game using geotagged photos. pages 300–303.

Vlachos, M., Gunopulos, D., and Kollios, G. (2002a). Robust similarity measures for mobile object trajectories. In *Database and Expert Systems Applications, 2002. Proceedings. 13th International Workshop on*, pages 721–726. IEEE.

Vlachos, M., Kollios, G., and Gunopulos, D. (2002b). Discovering similar multidimensional trajectories. In *Data Engineering, 2002. Proceedings. 18th International Conference on*, pages 673–684. IEEE.

Waga, K., Tabarcea, A., Chen, M., and Fränti, P. (2012). Detecting movement type by route segmentation and classification. In *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2012 8th International Conference on*, pages 508–513. IEEE.

Waga, K., Tabarcea, A., and Fränti, P. (2011). Context aware recommendation of location-based data. In *System Theory, Control, and Computing (ICSTCC), 2011 15th International Conference on*, pages 1–6. IEEE.

Waga, K., Tabarcea, A., Mariescu-Istodor, R., and Fränti, P. (2013). Real time access to multiple GPS tracks. pages 293–299.

Worsley, K. J. and Friston, K. J. (1995). Analysis of fMRI time-series revisitedagain. *Neuroimage*, 2(3):173–181.

Yanagisawa, Y., Akahani, J.-i., and Satoh, T. (2003). Shape-based similarity query for trajectory of mobile objects. In *Mobile data management*, pages 63–77. Springer.