Taylor & Francis
Taylor & Francis Group

Journal of Location Based Services

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tlbs20

Context-aware similarity of GPS trajectories

Radu Mariescu-Istodor & Pasi Fränti

To cite this article: Radu Mariescu-Istodor & Pasi Fränti (2020) Context-aware similarity of GPS trajectories, Journal of Location Based Services, 14:4, 231-251, DOI: 10.1080/17489725.2020.1842923

To link to this article: https://doi.org/10.1080/17489725.2020.1842923

Published online: 04 Nov 2020.

JOURNAL OF LOCATION BASED SERVICES
2020, VOL. 14, NO. 4, 231–251
https://doi.org/10.1080/17489725.2020.1842923

Taylor & Francis
Taylor & Francis Group

Context-aware similarity of GPS trajectories

Radu Mariescu-Istodor and Pasi Fränti

University of Eastern Finland, Kuopio, Finland

ABSTRACT

Measuring similarity of GPS trajectories has attracted a lot of attention in recent years. As a result, multiple trajectory similarity measures have been developed and are used in a wide set of applications which aim to extract meaningful information from large collections. In this paper, we focus on some of the most popular measures and study how they all can be adapted to use contextual information. We experiment using the buildings in an urban setting as the context and demonstrate how it impacts the similarity values. Experiments show that routes rank differently in terms of similarity in the presence of context which can have serious implications in applications such as trajectory search and clustering similar trajectories.

ARTICLE HISTORY
Received 7 August 2020
Accepted 23 October 2020

KEYWORDS
Trajectory similarity; context; GPS trajectories; similarity measures

1. Introduction

In recent years, GPS technology has become widely available in smart devices: phones, tablets, and watches. The wide availability of GPS-enabled devices makes it possible to collect large amount of location-based data. Such data includes geo-tagged photos, videos, and trajectories. Users record trajectories because of work: taxi, bus and truck drivers, train engineers, airplane pilots, or simply just for the pleasure to update a travel diary or sports tracking. Scientists track also animals and meteorological phenomena. As a result, the amount of location data is overwhelming and growing.

An important problem in understanding large amounts of trajectories is how to measure their similarity. Knowing the similarity serves as a foundation for many advanced analyses such as anomaly detection, clustering, classification, user similarity, and search (Shang et al. 2012; Wang and Liu 2012; Yanagisawa, Akahani, and Satoh 2003; Ying et al. 2010). Unlike similarity of single points, it is not obvious how to calculate the similarity of trajectories because they consist of multiple points, have high dimensionality and contain both spatial and temporal information. Multiple similarity measures have been introduced to serve various applications.

CONTACT Radu Mariescu-Istodor radum@cs.uef.fi

Some examples of similarity measures include longest common subsequence (Zheng and Zhou 2011), edit distance on real sequence (Lei Chen, Ozsu, and Oria 2005), edit distance with real penalty (Chen and Ng 2004), Euclidean distance (L_2 -norm) (Gradshteyn and Ryzhik 2000), dynamic time warping (Zheng and Zhou 2011), Fréchet distance (Eiter and Mannila 1994), Hausdorff distance (Rockafellar and Wets 2009), interpolated route distance (Trasarti et al. 2017) and cell similarity (Mariescu-Istodor and Fränti 2017).

However, none of these measures take into account the spatial context such as the existence of obstacles like rivers and buildings. This is needed in surveillance applications, where the visibility between the moving individuals is strongly preferred. It is also important in cleaning, snow removal, and other road maintenance tasks to mark down which parts of the road have been completed. Traditional methods may consider the trajectory by the main road to be the same as the one in the back yards because they are so close that a threshold-based method can easily fail in the presence of GPS errors.

In this paper, we present how to generalise existing measures to become context-aware. Figure 1 shows two examples of similar GPS trajectories. On the left, the similarity is measured by a traditional method. On the right, a context-aware method is used which considers the buildings as obstacles. The two trajectories on the top are on different sides of the buildings while there are no buildings in-between on the bottom. All traditional similarity measures would rank the trajectories in the top example more similar because of being closer. However, if we take into account the existence of the buildings, the similarity would drop dramatically.

In this paper, we exemplify using *buildings* as the context. We do this because it is easiest to find trajectories and buildings data in urban areas. We obtain building geometries (shapes) from OpenStreetMap.[1] Even though we focus on the buildings, the proposed approach generalises to other contextual information as well. One example can be to use *topography*: a boat trajectory on a river vs. a pedestrian trajectory by the river (see Figure 2). It is also important to use topography in applications like road network extraction from GPS trajectories (Mariescu-Istodor and Fränti 2018; Cao and Krumm 2009; Fathi and Krumm 2010; Chen and Cheng 2008; Edelkamp and Schrödl 2003). Otherwise, samples on land and on water may average into one trajectory on the side of the river (Yang, Mariescu-Istodor, and Fränti 2019), which would not be representative either of the two cases.

In addition to road network extraction, context-aware measure can also benefit applications like route recommendation (Dai et al. 2015; Kurashima et al. 2010; Waga et al. 2012a) and map matching (Lou et al. 2009; Brakatsoulas et al. 2005). All these applications strongly depend on the chosen trajectory similarity measure and can provide erroneous result if the context is not taken into account. For example, when a driving assistant displays the car on the nearest road, instead of a parking place.

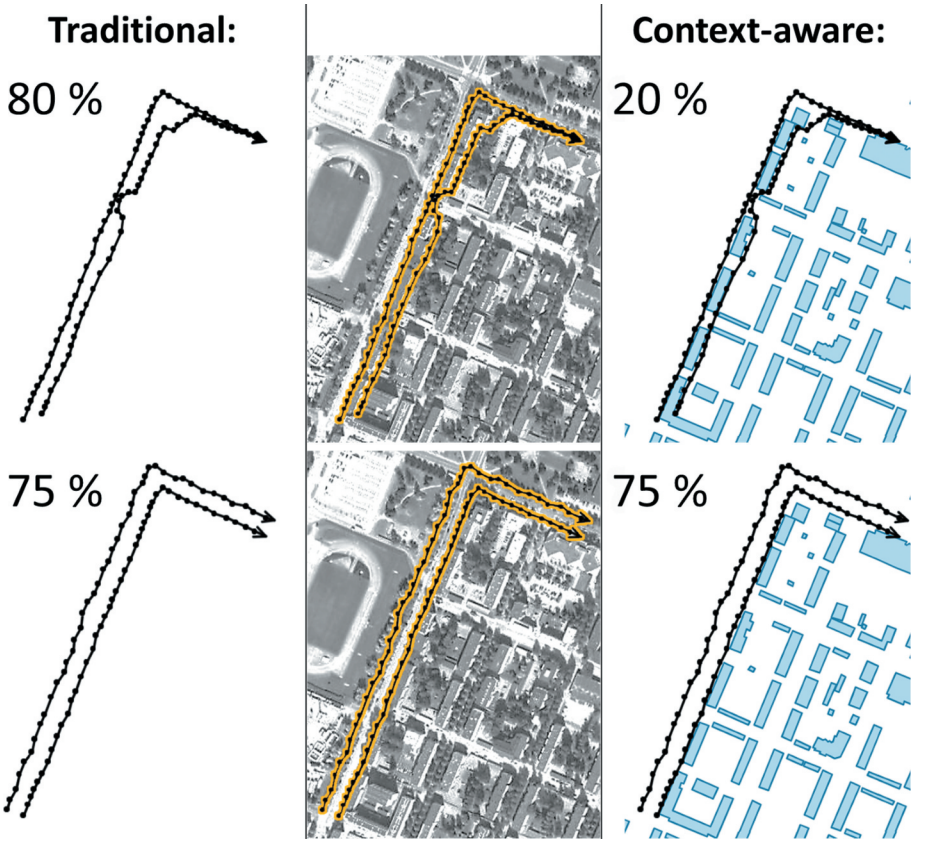


Figure 1. Two samples of GPS trajectories with a high similarity score (left). The similarity changes when context is used and buildings are considered as obstacles (right).

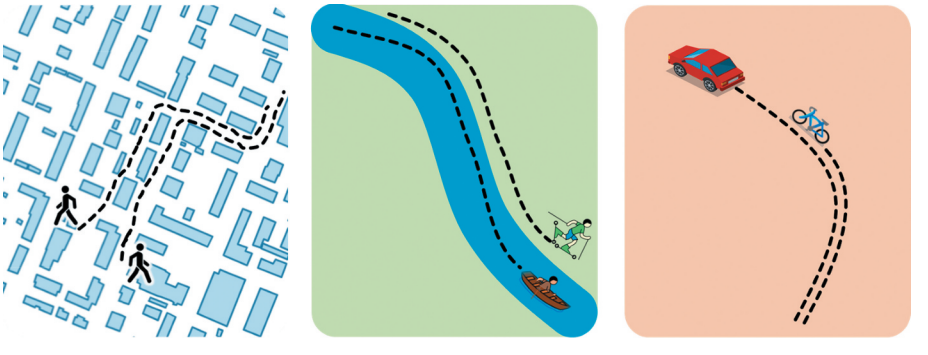


Figure 2. Three examples where the context affects the similarity. From left to right: buildings in an urban setting, moving on land vs. moving on water, and using different transportation modes.

A third example that uses *semantics* as the context is the detection of transportation mode (Waga et al. 2012b). Using this type of context can be helpful in the scope of road network extraction. Otherwise, the two trajectories

would merge, as they are spatially similar, and the expected movement speed will not be a representative one.

2. Trajectory similarity measures

A trajectory is an ordered set of GPS points and their respective timestamps. To calculate the similarity of trajectories, measures have been adapted from other fields such as longest common subsequence (Zheng and Zhou 2011) and edit distance-based measures (Lei Chen, Ozsu, and Oria 2005) from string similarity, Euclidean (Gradshteyn and Ryzhik 2000) and dynamic time warping based measures (Zheng and Zhou 2011) from time series analysis, and Fréchet (Eiter and Mannila 1994) and Hausdorff distances (Rockafellar and Wets 2009) from functional analysis. More recently, similarity measures specifically targeted to GPS sequences have also been introduced such as cell similarity (Mariescu-Istodor and Fränti 2017) and interpolated route distance (Trasarti et al. 2017). These last two measures take better into account the sparsity of the points than the measures adopted form other application areas.

To take the context into account, we present two alternative approaches: *Visibility* (VIS) and *Shortest Path* (SP), which are demonstrated in Figure 3. In the first variant, VIS, points obscured by some context are considered further than the Euclidean distance would imply. We determine that two points are obscured if their line of sight is interrupted because of the context. To conclude this, we compute the intersections of the line of sight with all segments from the building boundaries in the region. If a single intersection exists, the two points are obscured and the actual distance is weighed by a constant factor; or set it to infinite depending on the application.

The second variant, SP, is parameter-free. We compute the shortest path that avoids the context. In Figure 3, the path avoids the buildings in the shortest possible way. In practice, we first compute the visibility graph using Lee's algorithm (Lee

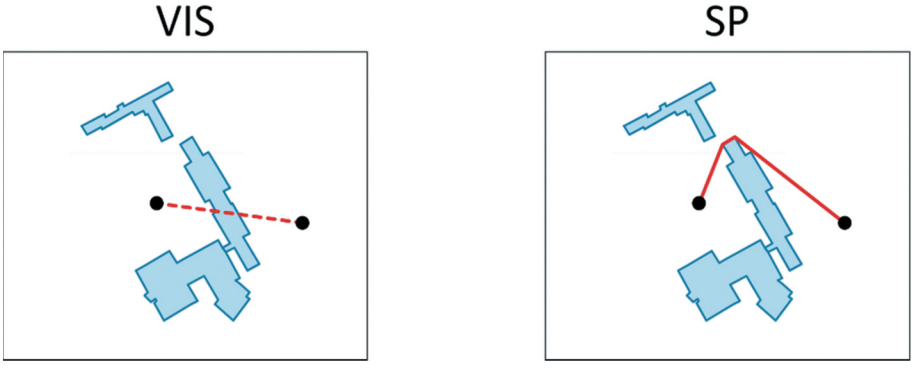


Figure 3. The way visibility (VIS) and shortest path (SP) variants handle distance calculations when buildings interfere with the line of sight.

1978) and then compute the shortest path in this graph using Dijkstra's algorithm (Cormen 2009). This variant has applications in orienteering (Fränti, Mariescu-Istodor, and Sengupta 2017), where participants do not necessarily stick to paths defined by a road network. Other applications are surveillance and rescue missions, where someone travelling on one trajectory needs to reach the other to provide some sort of assistance.

In the following, we describe how conceptually different similarity measures behave and how context-aware counterparts can be defined for each of them. We also highlight potential applications.

2.1. Similarity measures adapted from string processing

The *longest common subsequence* (LCSS) is a traditional measure for string similarity. Consider two strings: **M**O**PS**I and **M**A**PS**, their longest common subsequence is **MPS**, which has a length of three. We note that, unlike substrings, subsequences are not required to occupy consecutive positions in the two sequences. The longest common subsequence counts the number of matched characters from one string to another. The similarity is higher if more characters are matched. To compute a similarity score, the number of matches is divided by the length of the maximum of the two strings.

LCSS has been adapted for GPS trajectories in (Vlachos, Gunopulos, and Kollios 2002; Zheng and Zhou 2011) with the modification that two points may be matched if the distance between them is less than a constant ε, with a recursive formula given in Equation (1). In practice, the formula is implemented using dynamic programming with quadratic time complexity.

$$\mathrm{LCSS}(A,B) = \begin{cases} 0 & \text{, if } n = 0 \text{ or } m = 0 \\ 1 + \mathrm{LCSS}(\mathrm{Rest}(A), \mathrm{Rest}(B)) & \text{, if } d(\mathrm{Head}(A), \mathrm{Head}(B)) \leq \varepsilon \\ \max \begin{cases} \mathrm{LCSS}(\mathrm{Rest}(A), B) \\ \mathrm{LCSS}(A, \mathrm{Rest}(B)) \end{cases} & \text{, otherwise} \end{cases} \quad (1)$$

LCSS measure is suitable for trajectories when they are required to have points near each other (closer than ε). The measure tells how many points from one trajectory are matched to the other. The method handles point shifting well (Wang et al. 2013; Mariescu-Istodor and Fränti 2017) because points affected by large amount of noise are simply omitted from the matching: and the distance they shift is not taken into consideration.

The distance threshold ε is usually set between 10 and 60 m (Wang et al. 2013). In the latter, it is possible that one or even multiple buildings lie in-between as it can be seen in Figure 4. The drop in similarity is more significant in the VIS variant because in case of obscured points, weighted distances exceed the ε threshold.

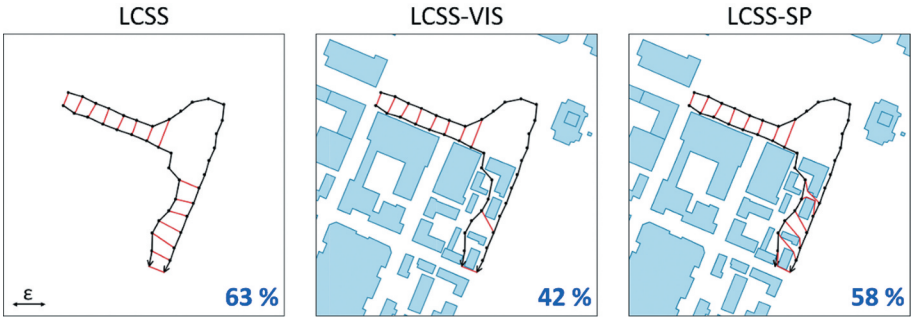


Figure 4. Two trajectories and LCSS similarity shown with and without contextual influence. The VIS and SP variants are both explained. The VIS variant parameter weights Euclidean distances by a factor of two between obscured points.

The SP variant is affected less because the obstacles are not too large and the shortest path around them often remains less than ε.

Another string inspired measure applied for GPS trajectories is *edit distance*, which counts the minimum number of characters that must be removed, added or substituted to transform one string into the other. For example, MOPSI may be transformed into MAPS with two operations: substituting the O into an A and removing the letter I. Two GPS trajectory similarity measures were invented based on this classical string definition: *edit distance on real sequence* (EDR) (Lei Chen, Ozsu, and Oria 2005) and *edit distance with real penalty* (ERP) (Chen and Ng 2004). In both versions, points are added, removed, or relocated to transform one trajectory into the other. Points within an ε distance are considered to be the same. ERP is a metric variant of EDR which was developed to allow efficient pruning techniques in spatial databases. Both these variants are of similar time complexity and behave similarly to LCSS (Wang et al. 2013; Mariescu-Istodor and Fränti 2017).

Other methods consider a user-defined matching thresholds. One of these is CATS (Hung, Peng, and Lee 2015), which instead of binary values for each pair of points (matched or not) uses a function that considers the Euclidean distance normalised over the ε threshold. EDwP (Ranu et al. 2015) projects points of the denser trajectory on the interpolated lines of the other trajectory using uniform distribution. One more recent related measure is MSM (Furtado et al. 2016), which also allows for partial matchings and many-to-many matchings.

Handling the context can be done similarly for all aforementioned methods and we will not study them separately in this paper. Because these methods impose a threshold to categorise points as similar or not, they can be applied in applications where travelling between the two trajectories is critical and needs to happen below a specified time. In this case, using the context is recommended to obtain more realistic values than the traditional measures would provide.

2.2. Similarity measures adapted from time series processing

Euclidean measure (Gradshteyn and Ryzhik 2000) compares the points at the same time instance. It is a naïve approach, in which every offset has a cumulative effect. It is, however, faster than any other similarity measures and is easy to implement. It is therefore a preferred measure when large amount of calculations are required; for instance in clustering. Unlike the methods adapted from string similarity, here, the actual distance between the points matters more. The farther the matched points, the lower the similarity will become. There is no commonly agreed method for converting the distance to a similarity value. However, at least in clustering application we can use the distance value as such. We will therefore show how the distance varies instead of the similarity.

$$\text{Euclidean}(A,B) = \sqrt{\sum_{i=1}^{\min(n,m)} d(a_i,b_i)^2} \tag{2}$$

The similarity of the trajectories is inversely proportional to the distance (see Figure 5). For the VIS variant, distance almost doubles from the non-contextual variant mostly due to the misalignment in the later part of the trajectories. The SP variant is not so different because, again, the buildings are not very large and can be avoided rather easily. The VIS variant may have applications such as concluding if a suspect was out of sight at any point during a chase.

Dynamic time warping (DTW) (Berndt and Clifford 1994; Zheng and Zhou 2011) allows for time dilation and for a single point to be matched to multiple others with the objective to minimise the total distance. The method is typically used when some sort of averaging is required (Hautamäki, Nykänen, and Fränti 2008) or when the application is to extract road networks (Mariescu-Istodor and Fränti 2018). DTW is implemented by dynamic

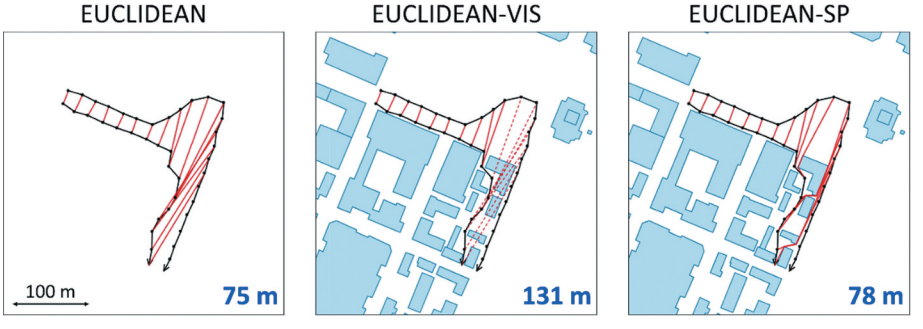


Figure 5. Two trajectories and Euclidean distance shown with and without contextual influence. The distance values shown are the average length of the distances between every pair of matched points. The VIS variant parameter is set to scale obstructed point distances by a factor of two.

programming which has quadratic time complexity (see Equation 3). Faster approximate variants have been also introduced (Salvador and Chan 2004). Recursive formula is as follows:

$$\mathrm{DTW}(A,B) = \begin{cases} 0 & \text{, if } n = m = 0 \\ \infty & \text{, if } n = 0 \text{ or } m = 0 \\ \mathrm{d}(\mathrm{Head}(\mathrm{A}), \mathrm{Head}(\mathrm{B})) + \min \begin{cases} \mathrm{DTW}(A, \mathrm{Rest}(B)) \\ \mathrm{DTW}(\mathrm{Rest}(A), B) \\ \mathrm{DTW}(\mathrm{Rest}(A), \mathrm{Rest}(B) \end{cases} & \text{, otherwise} \end{cases} \tag{3}$$

Figure 6 shows how the DTW distance is smaller than Euclidean due to the optimum matching of points. The large difference when using the VIS variant is not as large as the Euclidean case also because of the alignment. DTW may be used to group together trajectories into clusters and average them into segments for the purpose of road network extraction (Mariescu-Istodor and Fränti 2018). If the number of clusters is not specified, automatic methods may fail to detect it correctly and cause erroneous segments as seen in Figure 7. Adding the context increases the chance of correct detection for the number of clusters because of the increased distances between the trajectories recorded on the different streets, which have buildings in-between.

Interpolated route distance (IRD) is a recent similarity measure proposed in (R. Trasarti et al. 2017). It is similar to the DTW measure but it has two advantages. First, it uses interpolation to improve behaviour in case of different sampling intervals. Second, the algorithm provided in the paper works in linear time, which can be a significant advantage in many practical applications. Figure 8 shows that the method has similar behaviour to that of DTW; however, the distances are smaller due to the multiple added matches coming from the interpolation process.

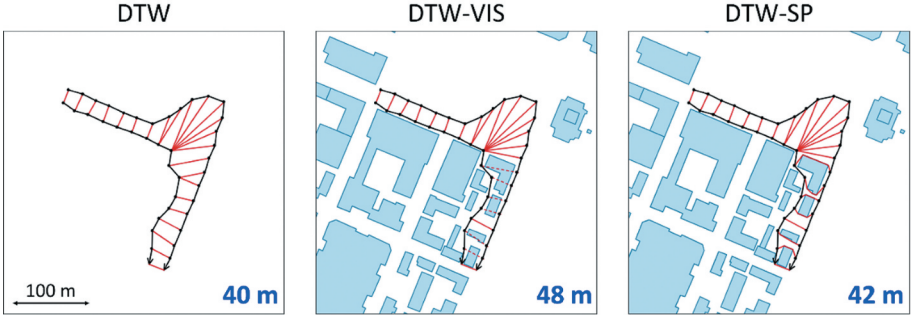


Figure 6. Two trajectories and DTW distance shown with and without contextual influence. The distance values shown are the average length of the distances between every pair of matched points. The VIS parameter is set to two.

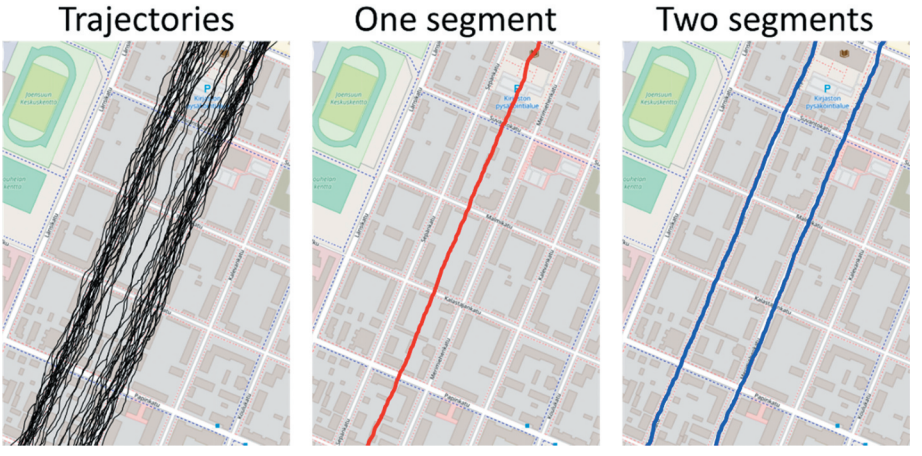


Figure 7. A set of similar trajectories averaged into one representative segment and two representative segments respectively.

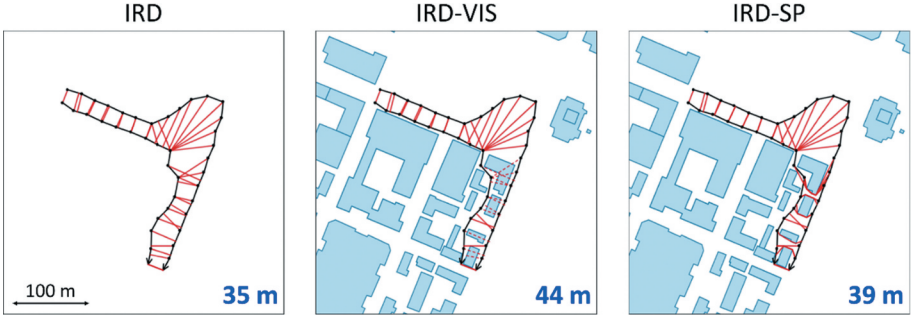


Figure 8. Two trajectories and IRD distance shown with and without contextual influence. The distance values shown are the average length of the distances between every pair of matched points. The VIS parameter is set to two.

2.3. *Fréchet*

The Fréchet distance was originally defined by Fréchet (1906) as a measure of distance between two curves. It was used to identify the geometrical similarity between curves. Algorithms to compute the distance are given in (Alt and Godau 1995) but they are all less efficient than computing the discrete Fréchet distance as proposed later by (Eiter and Mannila 1994). This variant is now the most widely used measure for GPS trajectories. The discrete Fréchet distance can be calculated using Equation 4, and it can be implemented using dynamic programming in quadratic time.

$$D_{Fréchet}(A_i, B_i) = max\begin{cases} \min(D(i-1,j), D(i-1,j-1), D(i,j-1)) \\ d(A_i, B_i) \end{cases} \quad (4)$$

Intuitively, the distance is the minimum possible length of a leash required to walk a dog, if the owner walks on one trajectory and the dog on the other, without allowing to backtrack.

Figure 9 shows how Fréchet changes when context is added. Fréchet is usually used in map matching (Lou et al. 2009; Brakatsoulas et al. 2005) and applying the context may help to match correctly in the case when there are buildings in between. Hausdorff distance (Rockafellar and Wets 2009) is similar to Fréchet but the direction of travel is not enforced. It behaves similarly in the presence of context. It is typically used in clustering applications (Chen et al. 2011) but due to the similar behaviour with Fréchet we will not discuss it further in this paper.

With the dog-owner example in mind, a third context-aware variant can be developed for the Fréchet distance, which requires the leash to be long enough so that the owner and the dog can reunite after surrounding a building. This distance may be calculated using the convex hull of the buildings as shown in Figure 10. However, possible application for this variant is unclear, so we consider it merely as a theoretical example.

More recently, several other methods were proposed for raw trajectory similarity. In (Ding, Trajcevski, and Scheuermann 2008), the proposed method, wDF,

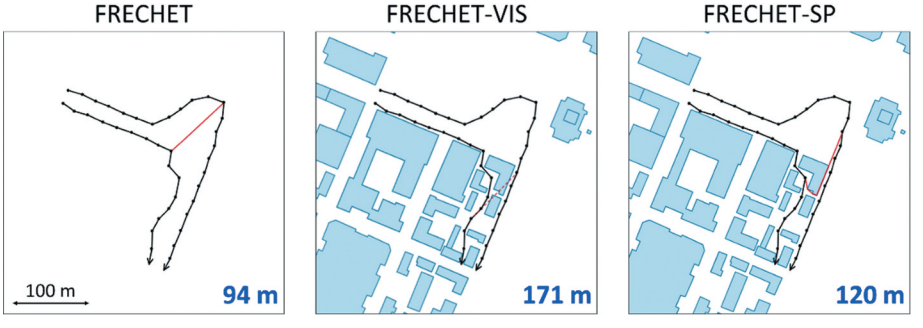


Figure 9. Two trajectories and Fréchet distance shown with and without contextual influence. The VIS parameter is set to two.

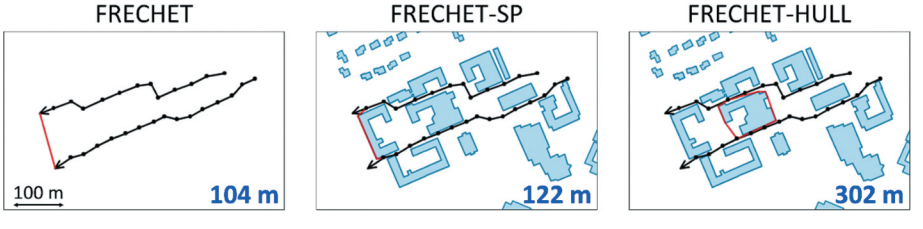


Figure 10. A secondary example for the Fréchet distance where the context is handled using shortest path and another convex hull – based variant (HULL) is introduced. The VIS parameter is set to two.

adapts the discrete Fréchet distance to consider only the pairs of points that are within a given time window. Another variant is presented in (Buchin and Purves 2013), where instead of computing distances between points, they are computed over a set of space–time prisms generated over the sampled trajectory. These other methods can be adapted to use context similarly as well.

From an application point of view, the Fréchet distance and its variants are useful when the maximum distance on two trajectories must not exceed a certain value. This can be the case when using walkie-talkies having a maximum range. Then, context such as buildings or dense forest can impact this maximum range and should be considered.

2.4. Cell similarity

Cell similarity (CSIM) is a recently introduced similarity measure that considered merely the traversed area by the two trajectories. It uses a grid to compute a cell representation for the two trajectories, and then similar to the Jaccard set-matching coefficient, it measures how many cells are in common relative to the total number of distinct cells. To compensate for the arbitrary division of a grid, which may allow points that are even 1 mm away to lie in different cells, CSIM uses morphological dilation with a square (3 x 3) structural element (see Figure 11).

The main advantages of CSIM are that the algorithm has linear time complexity and the result is not affected by point offset. The formula of CSIM is essentially the Jaccard coefficient modified to handle the dilated cells:

$$S(C_A, C_B) = \frac{\left|(C_A \cap C_B) \cup (C_A \cap C_B^d) \cup (C_B \cap {}^C_A d)\right|}{|C_A \cup C_B|} \tag{5}$$

To adapt CSIM to work with the context, we take every intersection cell and check every point inside. We consider all matches to every point of the other trajectory within the cell and the 8-cell dilated region. The VIS variant checks

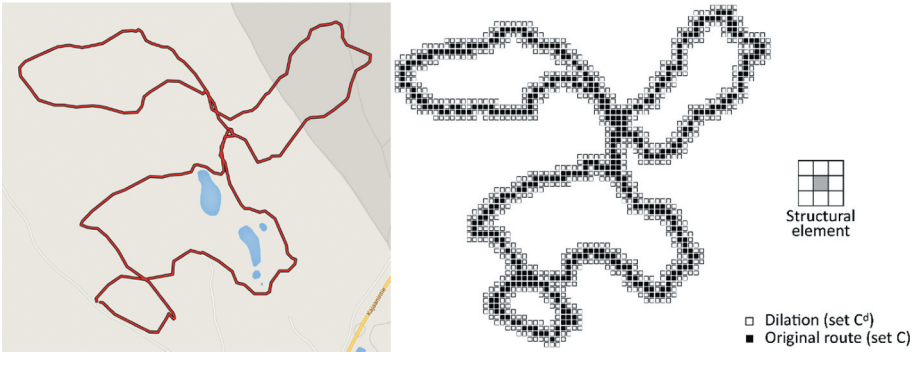


Figure 11. A sample route (left) and the cell representation with cell size 25x25 metres (right).

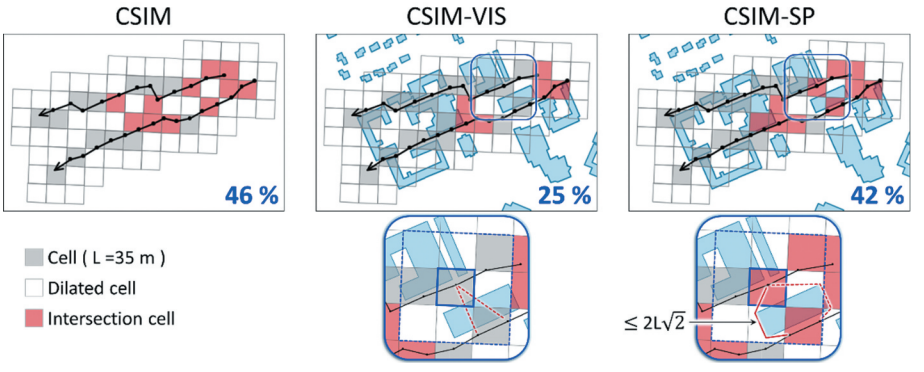


Figure 12. Two trajectories with 46% CSIM similarity. Context-aware variants vary significantly. The VIS variant 25% implies that there are many obstructions along the way. The SP variant is high (comparable to original CSIM) implying that only few of the obstructions are large. Below each variant we see how an intersection cell changes status in the VIS variant, but keeps it in the SP variant because the obstruction is not so large. The VIS parameter is set to two.

that at least one match is not obscured. If all matches are obscured, and the weighted point-to-point distances exceeds $2L\sqrt{2}$ the cell is no longer marked as an intersection cell. The $2L\sqrt{2}$ threshold is used because it is the maximum possible distance of two points in neighbouring cells can have. The SP variant computes the shortest path for all matches. If at least one has length less than or equal to $2L\sqrt{2}$, the cell remains classified as an intersection cell; otherwise, it loses its status (see Figure 12).

C-SIM is useful when the surface area covered is important. This includes tasks like searching for a lost object or marking down maintenance progress of roads, or cross-country skiing tracks. Using the context helps in dense regions where two roads are near each other but separated by narrow buildings or green area, for example.

3. Experiments

In our experiments, we consider the following four conceptually different similarity measures:

- Longest common subsequence (LCSS)
- Dynamic time warping (DTW)
- Frechet,
- Cell similarity (CSIM).

We will evaluate the trajectory similarity measure with and without the context support. In our experiments, we use the visual variant with the multiplication factor of 2.

3.1. The number of obstacles

In the first experiment, we investigate how the four measures behave in an artificial setting where we alter the number of obstructions in the context. To do this, we take two straight trajectories with the same rate for the points. The trajectories are parallel and at a distance of 10 metres from each other. We set the epsilon threshold for LCSS and the cell length for CSIM equal to 10 metres. We then add square shape obstructions randomly between the trajectories. The results are shown in Table 1.

We can see that the measures have quite different behaviour from each other. LCSS and DTW behave in a similar way, almost proportional to the number of obstructions inside the context. This can be useful for applications road network extraction where it is important to distinct between distances caused by errors GPS from distances caused by real obstacles like buildings or water channels. Fréchet distance becomes 20 metres immediately after adding even a single obstruction because it is blocking the visibility and the situation remains the same no matter how many more obstructions are added. This can be useful if it is vital for the application that these two tracking devices never get out of sight from each other. CSIM requires more obstructions to have an effect because the cells are more permissive than direct point-to-point matches.

3.2. Results with Mopsi data

In our second experiment, we use *Mopsi* data. Mopsi is a location-based social network created by the School of Computing from the University of Eastern

Table 1. Results of the experiment where we increase the number of obstructions.

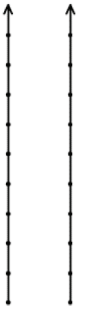
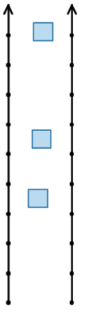
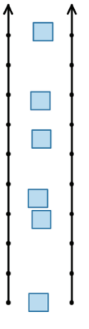
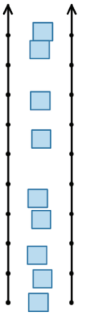
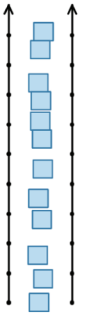
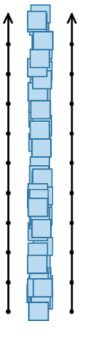
	0 Obstructions	3 Obstructions	6 Obstructions	9 Obstructions	12 Obstructions	∞ Obstructions
						
LCSS	100 %	91 %	64 %	55 %	45 %	0 %
DTW	10 m	11 m	14 m	15 m	16 m	20 m
Fréchet	10 m	20 m	20 m	20 m	20 m	20 m
CSIM	100 %	100 %	100 %	86 %	71 %	0 %

Table 2. Mopsi2014 dataset summary.

Trajectories	Points	Kilometres	Hours
6,779	7,850,387	87,851	4,504

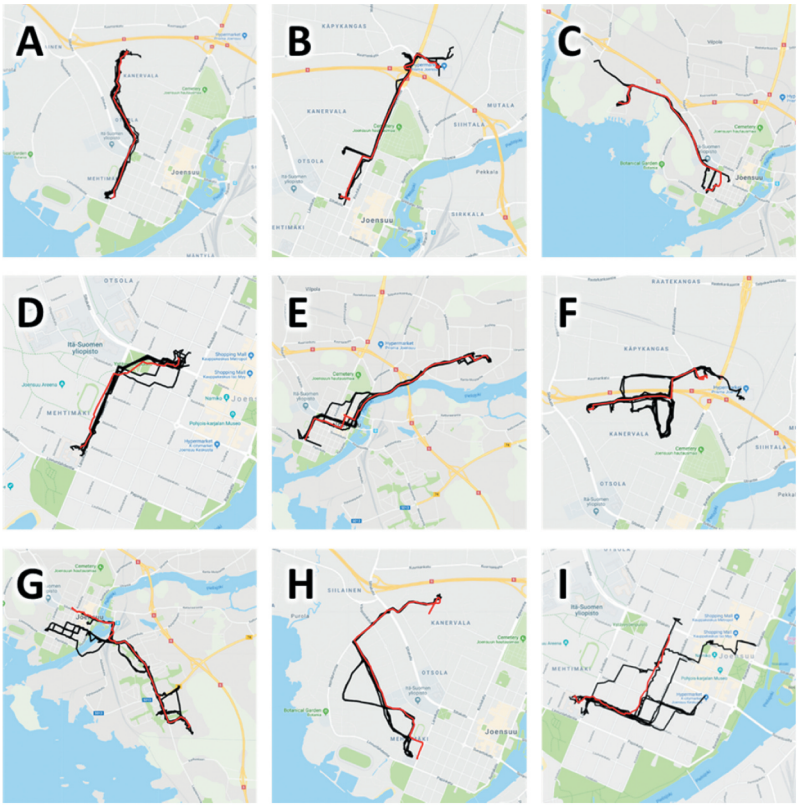


Figure 13. Nine sets of similar trajectories obtained by searching the database using a sample trajectory (shown in red colour).

Finland. Mopsi users can find out who or what is nearby. They can also track their movements, share photos and chat with friends. In our experiments, we use the Mopsi 2014 trajectory dataset[2] (Mariescu-Istodor and Fränti 2017), which is a subset of all trajectories in Mopsi database collected by the end of 2014. It contains 6,779 trajectories recorded by 51 users who have a minimum of 10 trajectories each. The trajectories consist of a wide range of activities including walking, cycling, hiking, jogging, orienteering, skiing, driving, travelling by bus, train, or boat. They exist on every continent except Antarctica. Most trajectories are in and around Joensuu, Finland. Table 2 summarises the Mopsi2014 dataset.

From this large dataset, we selected nine trajectories in Figure 13, which all have at least partial redundancy in the database (typically commuting from home to work or to store). For each of the nine trajectories, we find the most

similar trajectories from the database using the similarity ranking method described in (Mariescu-Istodor et al. 2014) and selected the top 10 that have the same move direction. The ranking was computed using CSIM, which does not consider direction; we therefore processed the result manually, and remove similar trajectories in opposite direction of travel to obtain easier to interpret results for the other measures. We then sorted these sets based on visual inspection, so that the ones with more variance are at the end. This ranking now represents the expected result (ground truth), and the correlation of the similarity ranking to the ground truth is expected to decrease from A to I. For each set, we calculate the similarity between the initial trajectory and all 10 similar ones using four conceptually different measures: LCSS, DTW, Fréchet, and C-SIM when using or not using the context.

We use the buildings in the region as the context. We obtained all buildings in the bounding box defined by every two trajectories using OpenStreetMap. We experiment using VIS variant with a multiplication factor of 2 and a threshold of 40 metres for measures that require such: LCSS and C-SIM. For each set we measure two things (see Table 3):

- Spearman's rank correlation (Corder and Foreman 2014) when context is/not used
- Difference (drop) in similarity when the context is added

In most cases, the correlation is less than 1, which means that when buildings are considered, the trajectories would be sorted in a different order. There is no clear evidence that the variance of the trajectories affect this order, however, lower correlations tend to exist for sets closer to city centre where more buildings are also present: Sets D, E, H, I. The order changes least when using

Table 3. Experiment results showing correlations (top) and at the bottom is the average similarity difference.

Set	LCSS	DTW	Fréchet	C-SIM
A	0.98	0.84	1.00	0.88
	0%	23 m	402 m	4%
B	0.98	0.99	1.00	0.99
	1%	92 m	465 m	1%
C	1.00	0.92	1.00	−0.15
	1%	60 m	548 m	9%
D	0.98	0.77	0.98	1.00
	2%	4 m	63 m	0%
E	0.98	0.89	1.00	1.00
	0%	34 m	265 m	1%
F	1.00	0.99	1.00	0.98
	1%	78 m	676 m	2%
G	0.98	1.00	1.00	0.82
	0%	43 m	695 m	3%
H	0.95	0.77	1.00	1.00
	1%	21 m	293 m	1%
I	0.79	0.98	1.00	0.98
	4%	90 m	479 m	1%

Figure 14. An example from set C where trajectories starting at the same location go on different sides of the buildings.

Fréchet or any other function based on the maximal difference. This is because on the location where the points are farthest from each other, there is a high probability for one or multiple buildings to exist in-between, meaning that similarities will decrease almost always and the order will remain unchanged. One remarkable situation happens on set C, when using C-SIM method: the correlation is negative, meaning the order is closer to being in reverse. This happens because of two reasons.

First, there are multiple trajectories that are nearly identical in the traditional sense. In this case, the order easily changes because of a region with high density of small buildings where multiple paths exist in-between, which users take (see Figure 14). The second reason is the 40 metre threshold required by C-SIM. If this value was set higher, the cells would be larger and nothing would obscure in this region. If we lowered the threshold slightly, LCSS would be affected similarly. It is less sensitive here because points are considered individually instead of the cells. In general, increasing the threshold makes the context-aware variants less affected by small obstructions.

When context influences the relationship between two trajectories, their similarity decreases (distance increases). In Table 3 we show also the average change between the reference trajectory and all others in the set. We note small values for LCSS and C-SIM because it is computed as the number of matched points and the distances themselves do not matter when over the threshold. This difference is

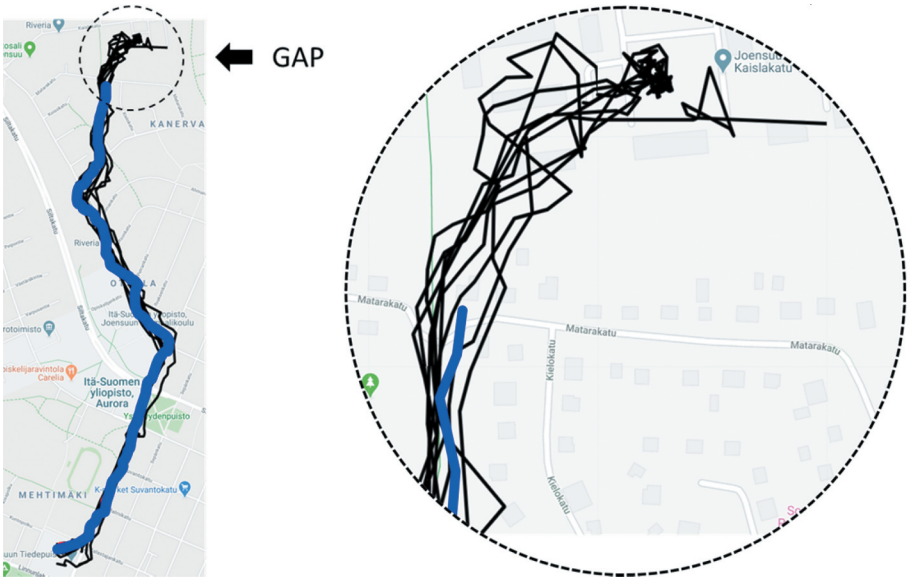


Figure 15. Set A and one trajectory highlighted (blue). The highlighted trajectory does not start at the same location as the others, affecting the similarity. Buildings exist in the region affecting context-aware similarity as well.

most significant for Fréchet, where the already large maximal differences are doubled every time they are obscured. We observe remarkably small differences for set D. This is because eight of the trajectories are nearly identical, and from the other two, one surrounds a park, thus, no buildings in-between, even though is seemingly different. Larger values for apparently more similar sets such as A appear because in A, not all trajectories start at the same location. Some are in fact included in the reference instead of perfectly matching it (see Figure 15). This causes a *gap* at the beginning or the end. If context exists in this region, a decrease in similarity (increase of distance) will appear.

The main limitation of the proposed measures is the need for the external data like the building database. While the proposed context-aware measures work as expected, the improvement can remain only marginal at dense areas. The extra complexity of the context-aware measure can be argued when accuracy is vital but if the goal is merely exploratory data analysis then it might not be worth it.

With the purpose of a more detailed analysis, we invite the reader to view the webpage[3] associated with this manuscript.

3.3. Processing time

The experiments described above completed in approximately 10 hours and we only experimented using the VIS variant because the SP variant would

have been too time consuming. Its main bottleneck is the visibility graph which has $O(N^2 \log(N))$ time complexity, where N is the total number of points of all building polygons in the region. Buildings tend to have complicated geometries often resulting in a few hundred vertices and links. After the visibility graph is formed, Dijkstra's algorithm runs on the graph with an additional $O(|E|+|V|\log|V|)$ where V is the number of vertices and E is the number of links (edges). These shortest paths need to be computed for every point pair of the two trajectories resulting in a quadratic number of shortest path computations.

To speedup these methods, the following improvements can be considered:

- Trajectory simplification via polygonal approximation (Chen, Xu, and Fränti 2012), which can dramatically reduce the number of point pairs.
- Spatial partitioning of the points (kd-trees, R-trees) to preprocess the buildings and query only those that interfere with the two input trajectories (in-between area) (Guttman 1984). This will form a *path of buildings* instead of a rectangular region which may reduce the number of buildings to the squared root of the original value.
- For the methods that use a threshold (LCSS, C-SIM), the traditional similarity can be computed as a preprocessing step. Then, the context needs to be considered only when the distance between a pair of points is lower than the threshold, reducing the number of buildings even further, which, in turn reduces the size of the visibility graph and the number of shortest path computations.

We did test these speedup methods but we believe they can reduce the processing time by several orders of magnitude. We left them as future work.

4. Conclusions

Trajectory similarity measure is a building block for many applications such as trajectory search, clustering, and map matching. Incorporating the context into the measure means that these applications can be easily upgraded by replacing the similarity measures by their context-aware variants without any other changes. We proposed two methods for adding contextual information to conceptually different trajectory similarity measures. We experimented on the most practical methods and the results indicate that all tested similarity measures provide different results when the context is considered. In addition, real-world GPS trajectories appear to rank differently when buildings are considered as the context within an urban environment. This implies that applications where trajectory similarity measure is a fundamental component are likely to benefit from using the context-aware variant.

Notes

1. https://www.openstreetmap.org.
2. http://cs.uef.fi/mopsi/routes/dataset.
3. http://cs.uef.fi/mopsi/routes/contextSimilarity.

Acknowledgments

We thank Mingyue Xie for the implementation of the interpolated route distance (IRD) method.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Pasi Fränti http://orcid.org/0000-0002-9554-2827

References

Alt, H., and M. Godau. 1995. "Computing the Fréchet Distance between Two Polygonal Curves." *International Journal of Computational Geometry & Applications* 5 (1n02): 75–91. doi:10.1142/S0218195995000064.

Berndt, D. J., and J. Clifford. 1994. "Using Dynamic Time Warping to Find Patterns in Time Series." *KDD Workshop* 10 (16): 359–370.

Brakatsoulas, S., D. Pfoser, R. Salas, and C. Wenk. 2005. "On Map-matching Vehicle Tracking Data." In Proceedings of the 31st international conference on Very large data bases, 853–864, Trondheim, Norway.

Buchin, M., and R. S. Purves. 2013. "Computing Similarity of Coarse and Irregular Trajectories Using Space-time Prisms." Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 456–459.

Cao, L., and J. Krumm. 2009. "From GPS Traces to a Routable Road Map." In Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems (ACM SIGSPATIAL GIS '09), Seattle, Washington, USA, 3–12.

Chen, C., and Y. Cheng. 2008. "Roads Digital Map Generation with Multi-track GPS Data. IEEE Int. Workshop on Education Technology and Training, 2008." *2008 International Workshop on Geoscience and Remote Sensing* 1: 508–511.

Chen, J., R. Wang, L. Liu, and J. Song. 2011. "Clustering of Trajectories Based on Hausdorff Distance." In Proceedings of the IEEE International Conference on Electronics, Communications and Control (ICECC '11), Ningbo, China, 1940–1944.

Chen, L., and R. Ng 2004. "On the Marriage of Lp-norms and Edit Distance." In Proceedings of the 30th International Conference on Very Large Data Bases-Volume (VLDB '04), Toronto, Canada, 792–803.

Chen, M., M. Xu, and P. Fränti. 2012. "A Fast Multiresolution Polygonal Approximation Algorithm for GPS Trajectory Simplification." *IEEE Transactions on Image Processing* 21 (5): 2770–2785. doi:10.1109/TIP.2012.2186146.

Corder, G. W., and D. I. Foreman. 2014. *Nonparametric Statistics: A Step-by-step Approach*, 142–144, John Wiley & Sons.

Cormen, T. H. 2009. *Introduction to Algorithms*. Cambridge, MA: MIT press

Dai, J., B. Yang, C. Guo, and Z. Ding. 2015. "Personalized Route Recommendation Using Big Trajectory Data." In IEEE 31st International Conference on Data Engineering, 543–554, Seoul, Korea.

Ding, H., G. Trajcevski, and P. Scheuermann. 2008. "Efficient Similarity Join of Large Sets of Moving Object Trajectories." 15th International Symposium on Temporal Representation and Reasoning, 79–87, NW Washington, DC.

Edelkamp, S., and S. Schrödl. 2003. "Route Planning and Map Inference with Global Positioning Traces." In *Computer Science in Perspective*, 128–151. Berlin, Heidelberg: Springer.

Eiter, T., and H. Mannila. 1994. *Computing Discrete Fréchet Distance. Tech. Report CD-TR 94/64*. Information Systems Department, Technical University of Vienna.

Fathi, A., and J. Krumm. 2010. "Detecting Road Intersections from Gps Traces." In Proceedings of the 6th International Conference on Geographic Information Science (GIScience '10), Zurich, Switzerland, 56–69.

Fränti, P., R. Mariescu-Istodor, and L. Sengupta. 2017. "O-Mopsi: Mobile Orienteering Game for Sightseeing, Exercising, and Education." *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 13 (4): 1–25. doi:10.1145/3115935.

Fréchet, M. M. 1906. "Sur Quelques Points Du Calcul Fonctionnel." *Rendiconti del Circolo Matematico di Palermo (1884–1940)* 22 (1): 1–72. doi:10.1007/BF03018603.

Furtado, A. S., D. Kopanaki, L. O. Alvares, and V. Bogorny. 2016. "Multidimensional Similarity Measuring for Semantic Trajectories." *Transactions in GIS* 20 (2): 280–298. doi:10.1111/tgis.12156.

Gradshteyn, I. S., and I. M. Ryzhik. 2000. *Tables of Integrals, Series, and Products*, 1114–1125. 6th ed. San Diego, CA: Academic Press.

Guttman, A. 1984. "R-trees: A Dynamic Index Structure for Spatial Searching." In Proceedings of the 1984 ACM SIGMOD international conference on Management of data (SIGMOD '84), New York, NY, USA, 47–57.

Hautamäki, V., P. Nykänen, and P. Fränti. 2008. "Time-series Clustering by Approximate Prototypes." IAPR International Conference on Pattern Recognition, Tampa, Florida, USA, 1–4.

Hung, C.-C., W.-C. Peng, and W.-C. Lee. 2015. "Clustering and Aggregating Clues of Trajectories for Mining Trajectory Patterns and Routes." *The VLDB Journal* 24 (2): 169–192. doi:10.1007/s00778-011-0262-6.

Kurashima, T., T. Iwata, G. Irie, and K. Fujimura. 2010. "Travel Route Recommendation Using Geotags in Photo Sharing Sites." In Proceedings of the 19th ACM international conference on Information and knowledge management, 579–588, New York, NY.

Lee, D.-T. 1978. "Proximity and Reachability in the Plane." PhD thesis, Champaign, IL, USA.

Lei Chen, M., T. Ozsu, and V. Oria. 2005. "Robust and Fast Similarity Search for Moving Object Trajectories." In Proceedings of the 2005 ACM SIGMOD international conference on Management of data and Symposium on Principles Database and Systems (SIGMOD/PODS '05), Baltimore, MD, USA, 491–502.

Lou, Y., C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang. 2009. "Map-matching for Low-sampling-rate GPS Trajectories." In Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems, 352–361, New York, NY.

Mariescu-Istodor, R., A. Tabarcea, R. Saeidi, and P. Fränti. 2014. "Low Complexity Spatial Similarity Measure of GPS Trajectories." In Proceedings of the 10th International Conference on Web Information Systems and Technologies (WEBIST'14), Barcelona, Spain, 62–69.

Mariescu-Istodor, R., and P. Fränti. 2017. "Grid-based Method for GPS Route Analysis for Retrieval." *ACM Transactions on Spatial Algorithms and Systems (TSAS)* 3 (3): 8.

Mariescu-Istodor, R., and P. Fränti. 2018. "Cellnet: Inferring Road Networks from Gps Trajectories." *ACM Transactions on Spatial Algorithms and Systems (TSAS)* 4 (3): 8.

Ranu, S., P. Deepak, A. D. Telang, P. Deshpande, and S. Raghavan. 2015. "Indexing and Matching Trajectories under Inconsistent Sampling Rates." IEEE 31st International Conference on Data Engineering,999–1010, Seoul, Korea.

Rockafellar, T. R., and R. J.-B. Wets. 2009. *Variational Analysis*. Vol. 317. Berlin, Germany: Springer Science & Business Media.

Salvador, S., and P. Chan. 2004. "FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space." In Prcoeedings of the 10th ACM International Conference on Knowledge Discovery and Data Mining Workshop on Mining Temporal and Sequential Data (SIGKDD '04), Seatle, Washington, USA, 70–80.

Shang, S., R. Ding, B. Yuan, K. Xie, K. Zheng, and P. Kalnis. 2012. "User Oriented Trajectory Search for Trip Recommendation." In Proceedings of the 15th ACM International Conference on Extending Database Technology, Berlin, Germany, 156–167.

Trasarti, R., R. Guidotti, A. Monreale, and F. Giannotti. 2017. "Myway: Location Prediction via Mobility Profiling." *Information Systems* 64: 350–367. doi:10.1016/j.is.2015.11.002.

Vlachos, M., D. Gunopulos, and G. Kollios. 2002. "Robust Similarity Measures for Mobile Object Trajectories. In Database and Expert Systems Applications, 2002." In Proceedings of the 13th IEEE International Workshop on Database and Expert Systems Applications (DEXA '02), Aix en Provence, France, 721–726.

Waga, K., A. Tabarcea, M. Chen, and P. Fränti. 2012b. "Detecting Movement Type by Route Segmentation and Classification." In 8th International Conference on Collaborative computing: networking, applications and worksharing (CollaborateCom), 508–513. IEEE: Pittsburgh, PA.

Waga, K., A. Tabarcea, and P. Franti. 2012a. "Recommendation of Points of Interest from User Generated Data Collection." In IEEE 8th International Conference on Collaborative Computing: Networking, Applications and Worksharing, 550–555. Pittsburgh, PA.

Wang, H., H. Su, K. Zheng, S. Sadiq, and X. Zhou. 2013. "An Effectiveness Study on Trajectory Similarity Measures." In Proceedings of the 24th Australasian Database Conference (ADC '13), Adelaide, Australia, 13–22.

Wang, H., and K. Liu. 2012. "User Oriented Trajectory Similarity Search." In Proceedings of the ACM SIGKDD International Workshop on Urban Computing (UrbComp '12), Beijing, China, 103–110.

Yanagisawa, Y., J.-I. Akahani, and T. Satoh. 2003. "Shape-based Similarity Query for Trajectory of Mobile Objects." In Proceedings of the 4th International Conference on Mobile Data Management (MDM '03), Melbourne, Australia, 63–77.

Yang, J., R. Mariescu-Istodor, and P. Fränti. 2019. "Three Rapid Methods for Averaging GPS Segments." *Applied Sciences* 9 (22): 4899. doi:10.3390/app9224899.

Ying, J. J.-C., E. H.-C. Lu, W.-C. Lee, T.-C. Weng, and V. S. Tseng. 2010. "Mining User Similarity from Semantic Trajectories." In Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks (ACM SIGSPATIAL GIS '10), San Jose, CA, USA, 19–26.

Zheng, Y., and X. Zhou. 2011. *Computing with Spatial Trajectories*. Springer Science & Business Media.