2016 IEEE International Conference on Robotics and Automation (ICRA)
Stockholm, Sweden, May 16-21, 2016

Unsupervised Trajectory Compression

Padraig Corcoran[1], Peter Mooney[2] and Guoquan Huang[3]

***Abstract*— We present a method for compressing trajectories in an unsupervised manner. Given a set of trajectories sampled from a space we construct a basis for compression whose elements correspond to paths in the space which are topologically distinct. This is achieved by computing a canonical representative for each element in a generating set for the first homology group and decomposing these representatives into a set of distinct paths. Trajectory compression is subsequently accomplished through representation in terms of this basis. Robustness with respect to outliers is achieved by only considering those elements of the first homology group which exist in the super-level sets of the *Kernel Density Estimation* (KDE) above a threshold. Robustness with respect to small scale topological artifacts is achieved by only considering those elements of the first homology group which exist for a sufficient range in the super-level sets. We demonstrate this approach to trajectory compression in the context of a large set of crowd-sourced GPS trajectories captured in the city of Chicago. On this set, the compression method achieves a mean geometrical accuracy of 108 meters with a compression ratio of over 12.**

I. INTRODUCTION

Trajectory data represents a fundamental component in a wide spectrum of application domains including robot path planning [1], [2], [3], traffic analysis [4] and social behaviour analysis [5]. Given the huge volumes of trajectory data being generated by such applications, the ability to perform effective compression is necessary in many cases. In many of these applications, the trajectories in question are captured in spaces where movement is constrained to a fixed set of paths which are topologically distinct. In the context of a two dimensional robot configuration space such paths correspond to, for example, going left or right around an object in the space [6], [1]. In the context of vehicles moving in a street network such paths correspond to individual streets. This discussion implies that one approach to compression is to determine such topologically distinct paths and use them as a basis for compression. Evidently this represents a lossy compression scheme where that lost corresponds to exact spatial and temporal information within individual paths or basis elements.

Using a set of paths as a basis for compression has previously been considered by many authors in the context of vehicles moving in a street network [7], [8], [9], [10]. These works are supervised in the sense that they assume that an accurate model or map of the network, and in turn that the basis used for compression, is known a priori. However in many cases a prior model of the space may not be available and even if it is, it may not be accurate and current. Therefore it would be of great value if such compression could be accomplished in an unsupervised manner without the requirement of a prior model. That is, where the paths or basis used for compression are inferred from the actual trajectory data to be compressed. Design of such a compression scheme represents the contribution of this paper.

Inferring a set of topologically distinct paths in a space given a set of trajectories sampled from that space represents a difficult problem. A trajectory is typically represented as a discrete sequence of points as opposed to a continuous function and one typically has only a sample of trajectories from which to infer the paths in question. Trajectories may be subject to the following two forms of noise which are topological in nature [11]. Firstly, the set of trajectories may contain outliers corresponding to trajectories which traverse insignificant paths not traversed frequently and therefore should not be included in the set of basis elements. For example, in the context of GPS trajectories significant paths correspond to those contained in a street network while insignificant paths correspond to random paths through a large open area. Secondly, points in a trajectory may exhibit localization error. For example, trajectories captured using a consumer GPS will exhibit localization error of roughly ten meters but this may be higher in urban areas due to signal attenuation. Such errors may result in the introduction of small scale topological artifacts or holes which may in turn be incorrectly identified as additional paths in the space. In the presence of both these forms of noise it is necessary to infer the set of paths in a robust manner.

To overcome these challenges we propose a novel solution to the problem of inferring a set of topologically distinct paths in a space. This solution is based on the insight that the paths in question correspond to a decomposition of the elements in the first homology group of the space. It contains the following steps. The super-level sets of the *Gaussian Kernel Density Estimation* (KDE) of the trajectory points are represented using a combinational representation known as an *upper star filtration* [12]. Using this representation we compute a canonical representative for each element in a generating set for the homology classes which correspond to cycles in the space. However, due to the fact that these cycles are not disjoint or localized [13] and trajectories generally correspond to point to point movement as opposed to cycles, they do not represent a suitable basis for compression. We therefore perform a decomposition of this generating set into a set of paths which are disjoint, localized and in turn form a more appropriate basis for compression. In order to

[1] Padraig Corcoran is with the School of Computer Science & Informatics, Cardiff University, Wales, UK. corcoranp@cardiff.ac.uk

[2] Peter Mooney is with the Department of Computer Science, Maynooth University, Maynooth, Ireland. peter.mooney@nuim.ie

[3] Guoquan Huang is with the Department of Mechanical Engineering, University of Delaware, USA. ghuang@udel.edu

978-1-4673-8026-3/16/$31.00 ©2016 IEEE

achieve robustness with respect to trajectory outliers, we only consider those elements in the generating set which exist in the super-level sets above a specified threshold. This solution draws from recent works in the area of robust topological inference [14], [15]. Moreover, in order to achieve robustness with respect to small scale topological artifacts or holes we only consider those elements of the generating set which exist in the super-level sets for a sufficient range. This solution draws from works in the area of persistent homology [16].

In this paper we focus on the problem of trajectory compression in the context of GPS trajectories. To illustrate the characteristics of such trajectories consider the set of GPS trajectories displayed in Figure 1(a) which form a running example in this paper. This set contains 892 individual trajectories captured through crowd-sourcing in the city of Chicago. It is evident from this figure that both forms of topological noise discussed above are evident. For example, a number of paths are only traversed by a single trajectory and therefore can be considered outliers. Despite this focus on GPS trajectories, the proposed methodology has the potential to be applied to other types of trajectories.

The layout of this paper is as follows. In section II we review related works on trajectory compression. Section III describes how the super-level sets of the Gaussian KDE are represented using an upper star filtration. Section IV introduces necessary concepts from homology theory. Section V describes the approach used to robustly compute a canonical representative for each element in a generating set for the homology classes. In section VI we describe the decomposition of these elements into a set of paths which form an appropriate basis for compression and how compression with respect to this basis is achieved. Finally in sections VII and VIII results and conclusions are presented respectively.

II. RELATED WORKS

As described in the preceding section, trajectory compression methods can broadly be categorized as supervised and unsupervised methods. In the following we briefly review related works in each of these categories.

The most widely used unsupervised compression method is that of Douglas-Peucker [17]. This method performs compression by selecting a subset of the original trajectory points such that the spatial disparity between the piecewise linear curves corresponding to the original and compressed trajectory is minimized. A variation of the Douglas-Peucker method was proposed by [18] which minimizes both spatial and temporal disparity. [19] performs compression by assigning semantic information, such as landmarks visited, to trajectory points and selecting a subset such that semantic disparity is minimized. [20] demonstrated the application of core-sets, which is a means of approximating a solution to an optimization problem, to trajectory compression. [21] proposed a method for sparsifying Probabilistic Roadmap Planner (PRM) roadmaps such that the resulting trajectories contain fewer points. An empirical evaluation of a number of unsupervised trajectory compression techniques was presented in [22].

All existing supervised trajectory compression methods perform compression by representing each trajectory in terms of a basis whose elements correspond to individual paths in a prior model of the space. In the context of GPS trajectories these elements typically correspond to individual street segments. We now briefly describe these methods. [23] presented four methods which use spatial distance, heading information, topological information and curve matching for determining a representation in terms of the basis. [7] proposed a method which considers spatial and temporal disparity when determining a representation in terms of the basis. [24] posed the problem in terms of performing inference with respect to a Hidden Markov Model. A number of authors have considered the Fréchet Distance as a means of determining a representation in terms of the basis [25].

Similar to the above supervised compression methods, the compression method proposed in this paper represents each trajectory in terms of a basis whose elements correspond to individual paths. However this basis is determined in a unsupervised manner *without* the requirement for a prior model of the space.

III. UPPER STAR FILTRATION

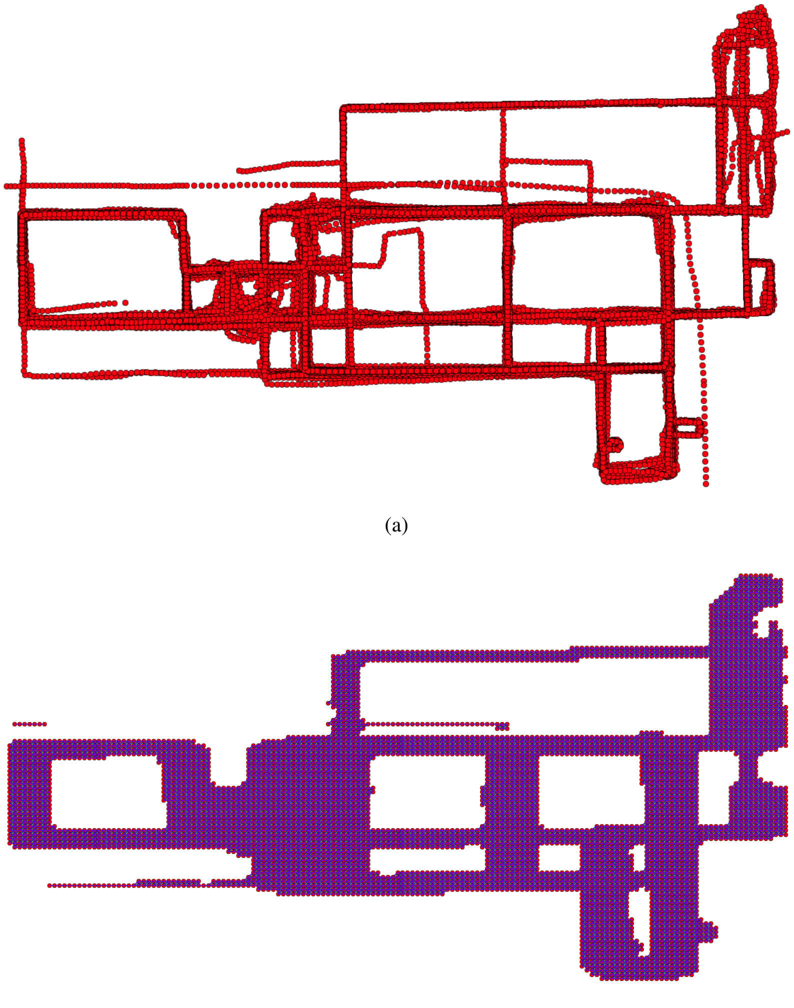
In this section we first describe the methodology used to compute a *Kernel Density Estimation* (KDE) of GPS trajectory points. We subsequently describe how the super-level sets of this density are represented using a combinatorial representation known as a filtration. This representation is used as a platform to robustly extract paths in the space.

Due to page constraints we do not review background on KDE but instead direct the interested reader to the following resource [26]. The KDE is computed using a Gaussian kernel over a grid of points [27], [14], [15]. Specifically we used a grid where the spacing between points is 25 meters and the bandwidth of the Gaussian kernel bandwidth is 0.05 meters. Empirically we found these values to be appropriate.

An (abstract) simplicial complex $\mathcal{K}$ is a finite set such that for each $\sigma \in \mathcal{K}$ all subsets of σ are also elements in $\mathcal{K}$. Each $\sigma \in \mathcal{K}$ is called a simplex or k-simplex where $|\sigma| = k + 1$ is the dimension of the simplex. The faces of a simplex σ correspond to all simplices τ where $\tau \subset \sigma$ [11], [28]. A function $f : \mathcal{K} \rightarrow \mathbb{R}$ is monotonic decreasing if $f(\sigma) \geq f(\tau)$ whenever σ is a face of τ [12]. This property implies that the super-level set $K(a) = f^{-1}[a, \infty)$ is a sub-complex of $\mathcal{K}$ for every $a \in \mathbb{R}$. If m is the number of simplices in $\mathcal{K}$ and f is monotonic decreasing, there exists a sequence of n values $a_1 > a_2 > \cdots > a_n$ such that given $a_0 = \infty$ and $\mathcal{K}_i = K(a_i)$, the sequence of complexes defined by Equation 1 exists. Such a sequence of complexes is known as a *filtration* of $\mathcal{K}$.

$$\emptyset = \mathcal{K}_0 \subseteq \mathcal{K}_1 \subseteq \cdots \subseteq \mathcal{K}_n = \mathcal{K} \tag{1}$$

In order to represent the super-level sets of the KDE as a filtration, we first construct a simplicial complex corresponding to the grid of points for which estimates were made. For


(a)

(b)

Fig. 1. A total of 118,221 GPS points corresponding to 892 individual trajectories captures by vehicles traversing the Chicago street network are displayed in (a). One of simplicial complexes in the upper star filtration of the Kernel Density Estimation is displayed in (b).

each grid point we construct a corresponding 0-simplex. For each pair of grid points which are horizontally, vertically or main diagonally adjacent we define a corresponding 1-simplex. For each triple of grid points where all subsets of pairs are horizontally, vertically or main diagonally adjacent we define a corresponding 2-simplex. This construction is illustrated in Figure 2.

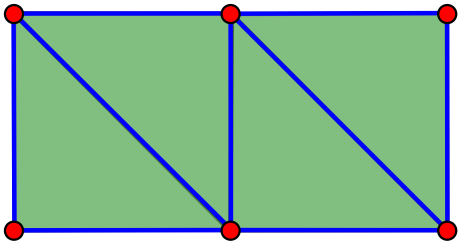


Fig. 2. For a gird of size 2×4 the corresponding simplicial complex is illustrated where red dots represent 0-simplices, blue lines represent 1-simplices and green triangles represent 2-simplices.

Toward constructing an appropriate filtration of this simplicial complex, we define the following monotonic decreasing function $f : \mathcal{K} \to \mathbb{R}$. If σ is a 0-simplex ($|\sigma| = 1$) of $\mathcal{K}$, $f(\sigma)$ evaluates to the density estimate at the corresponding grid point. If σ is a 1-simplex or 2-simplex we define $f(\sigma) = \max(f(\tau) : \tau \in \sigma, |\tau| = 1)$. For a given simplex σ we refer to $f(\sigma)$ as the *density* of σ. The filtration corresponding to the function f is known as an *upper star filtration* [12]. In practice this filtration is constructed by ordering the simplices in $\mathcal{K}$ such that simplices with higher density precede simplices of lower density. If two simplices have the same density we order them by dimension otherwise we order them arbitrary. We then define $\mathcal{K}_i$ to be that simplicial complex which is obtained by adding the i simplex σ_i in this ordering to $\mathcal{K}_{i-1}$. A simplicial complex corresponding to a super-level set of the trajectories in Figure 1(a) is displayed in Figure 1(b)

IV. HOMOLOGY THEORY

The purpose of this section is to provide an introduction to necessary background theory in homology; for greater details the reader is encouraged to consult the following text [29]. Intuitively homology is the study of cycles or holes in a space and is an abelianization of the fundamental group [30].

Let $\mathcal{K}$ be a simplicial complex. The mathematical structure c defined by Equation 2 is k-chain where each $\sigma_i \in \mathcal{K}$ is a k-simplex and each λ_i is an element of a given field. For the purposes of this work we consider the field $\mathbb{Z}_2$ as is common in path planning applications [31].

$$c = \sum \lambda_i \sigma_i \tag{2}$$

The vector space over all k-chains is called chain group $C_k(\mathcal{K})$. The boundary map which maps a k-simplex $\sigma = [v_1, \ldots, v_{k+1}]$ to the sum of its faces is defined by Equation 3 where $\hat{v}_i$ indicates the deletion of v_i from the sequence. It can be verified that $\partial_{k+1}\partial_k = 0$ [29]. Application of the boundary map to the chain groups forms a chain complex C_* as defined by Equation 4 [32].

$$\partial_k \sigma = \sum_{i=1}^{k+1} [v_1, \ldots, \hat{v}_i, \ldots, v_{k+1}] \tag{3}$$

$$\ldots \to C_{k+1}(\mathcal{K}) \xrightarrow{\partial_{k+1}} C_k(\mathcal{K}) \xrightarrow{\partial_k} C_{k-1}(\mathcal{K}) \to \ldots \tag{4}$$

A k-chain $c \in C_k(\mathcal{K})$ is a k-boundary if there exists some $d \in C_{k+1}(\mathcal{K})$ such that $c = \partial d$ and a p-cycle if $\partial c = 0$. The set of all k-boundaries and k-cycles are denoted by $B_k(\mathcal{K})$ and $Z_k(\mathcal{K})$ respectively. It is evident that both are subgroups of $C_k(\mathcal{K})$. As a consequence of $\partial_{k+1}\partial_k = 0$, $B_k(\mathcal{K}) \subseteq Z_k(\mathcal{K})$. The quotient group $H_k(\mathcal{K}) = Z_k(\mathcal{K})/B_k(\mathcal{K})$ is called the k-homology group of $\mathcal{K}$ and its rank is called the k^{th} Betti-number. Intuitively the k^{th} Betti-number equals the number of k-dimensional holes in $\mathcal{K}$. Using this intuition, the 0^{th} Betti-number corresponds to the number of connected components while the 1^{th} Betti-number corresponds to the number of one dimensional holes.

The equivalence class of a k-cycle in $H_k(\mathcal{K})$, denoted $[c]$, is called a homology class. Two p-cycles belonging to the same homology class are said to be homologous. We are principally interested in the homology classes of $H_1(\mathcal{K})$ because these correspond to trajectory cycles in a given space. Intuitively two cycles are homologous if the region they enclose contains no holes [33], [30]. As such we subsequently use the term homology classes when referring to homology classes of $H_1(\mathcal{K})$. In the case of $\mathbb{Z}_2$ coefficients, a homology group with rank n has order 2^n where the order represents the number of homology classes [12].

V. HOMOLOGY CLASSES

In this section we describe a method for robustly computing a canonical representative for each element in a generating set for the homology classes. These canonical representatives are subsequently used to extract a basis for compression in Section VI.

Consider the filtration described in Section III and the event of adding the simplex σ_i of dimension d_i to $\mathcal{K}_{i-1}$ to obtain $\mathcal{K}_i$. Upon this event, two possible changes of the homology can occur [34]. Either an element in the d_i homology group is created, in which case σ_i is called a *positive* simplex, or an element of the $d_i - 1$ homology group is destroyed, in which case is called σ_i a *negative* simplex. Each negative simplex σ_j has a corresponding positive simplex σ_i where $i < j$ such that σ_j destroys the element of the homology group created by σ_i. The pair (σ_i, σ_j) is called a *persistence pair*. A subset of positive simplices, which create an element of an homology group that is subsequently not destroyed, are not paired. These are called *essential simplices* [34]. Every simplex of $\mathcal{K}$ belongs either to a persistence pair or is essential. In this work the method of [35] is used to compute the persistence pairs and essential simplexes.

To compute a canonical representative for each element in a generating set for the homology classes we employ the method proposed by [36], [37]. We construct a spanning tree for $\mathcal{K}$ consisting of all negative 1-simplices. Adding each positive 1-simplex to this spanning tree creates a graph containing a single 1-cycle. This cycle is an element of a generating set of the group of canonical representatives sought. Given this generating set, a canonical representation for all elements of the homology classes can be subsequently obtained through application of the group operator. In the case of $\mathbb{Z}_2$ coefficients the group operator corresponds to exclusive or.

Each element in the generating set is assigned a birth and death density corresponding to the density at which it was created and destroyed respectively. The function used to assign density was introduced in section III. If an element is created but never destroyed it is assigned a death density of ∞. The *persistence* of an element is the range between its corresponding birth and death densities. In order to achieve robustness with respect to GPS localization error and in turn the introduction of small scale topological artifacts or holes we only consider those elements of the generating set with persistence above a specified threshold. This solution has been demonstrated to be effective and is part of a larger body of work in the area known as *persistent homology* [16]. In order to achieve robustness with respect to trajectory outliers, we only consider those elements of the generating set where the corresponding birth density is above a specified threshold.

VI. COMPRESSION BASIS & METHOD

In this section we present a method for computing a set of paths which form an appropriate basis for compression. We subsequently present a method for compressing a given trajectory by representing it in terms of this basis.

In the previous section we presented a method for computing a canonical representative for each element in a generating set for the homology classes. However these representations do not form a suitable basis for compression for two reasons. Firstly, while topologically correct, these representatives may not be localized geometrically and correspond cycles of long geometrical length around multiple holes in the space [13]. [13] proposed a method for generating representatives such that they are localized with respect to a given cover of the space. However specifying a suitable cover is challenging and would require some other form of prior knowledge about the space in question. Secondly, most trajectories correspond to point to point trajectories as opposed to cycles. As such, a set of cycles does not represent a suitable basis for compression. To overcome these issues, we propose a method for extracting a set of disjoint paths in the space where individual paths correspond to, for example, going left or right around a particular hole in the space.

The proposed method contains the following two steps. Firstly we take the *logical or* of all canonical representatives in the generating set. Next we identify the set of 0-simplices where multiple paths meet and in turn the set of individual paths between these 0-simplices. These paths form the basis which is used as a basis for compression. For example, consider again the set of trajectories displayed in Figure 1(a). Figure 3 displays the corresponding set of 0-simplices where multiple paths meet and the set of paths between these 0-simplices which are used as a basis for compression. It should be noted that in the context of GPS trajectories captured in a street network, these paths will generally correspond to individual streets in the network. As discussed in the related works section of this paper, individual streets are commonly used as a basis for compression in supervised compression schemes. As such, the ability to generate such a basis in an unsupervised manner represents one of the major benefits of the proposed compression methodology.

In order to compress a given trajectory by representing it in terms of the above basis the following method is employed. For each point in a given trajectory to be compressed we *match* it to the closest basis element in terms of Euclidean distance. A KD-tree is used to determine all such matches. A basis element is subsequently determined to be included in the trajectory representation if the number of points matched to that element divided by its geometrical length exceeds a specified threshold. Otherwise it is excluded.

VII. EXPERIMENTS

This section is structured as follows. In section VII-A we describe the data used within a set of experiments along with performance measures considered. In section VII-B we present the results of these experiments.

A. DATA AND PERFORMANCE MEASURES

Toward demonstrating the effectiveness of the proposed methodology for compressing a set of trajectories in an unsupervised manner we considered the set of GPS trajectories illustrated in Figure 1(a). This set contains 892 trajectories and the corresponding total number of GPS points is 118,221. The trajectories were captured in the city of Chicago through crowd-sourcing and were obtained from [38]. A UTM coordinate system, where distances measured are in meters, is used. The length and width of the axis aligned bounding box for these trajectories is 3,900 and 2,375 meters respectively.

The performance of the proposed compression method was measured with in terms of geometrical accuracy and data compression ratio. To measure geometrical accuracy with respect to a single trajectory the Hausdorff distance between the original representation of the trajectory and its corresponding compressed representation is evaluated. Given two subsets in a metric space, the Hausdorff distance measures the maximum distance of one subset to the nearest point in the other subset. To measure geometrical accuracy with respect to a set of trajectories, the mean and standard deviation of the Hausdorff distance between each individual trajectory and its corresponding compressed representation is evaluated.

Data compression ratio is a commonly used measure of compression ability and is equal to the ratio of uncompressed data size to compressed data size. We approximate each of these data sizes by the corresponding number of numerical values they contain.

B. RESULTS

To illustrate the compression results achieved consider Figure 4 which displays two distinct trajectories and their corresponding compressed representations. It is evident that the first trajectory in Figure 4(a) and its compressed representation are very similar. This is reflected by the corresponding low Hausdorff distance of 69 meters. The second trajectory is also similar to its compressed represented for the most part. However one of the boundaries of this trajectory lies in the center of a basis element and as such that boundary cannot be well represented in terms of the basis elements. This is reflected by the corresponding Hausdorff distance of 215 meters which is significantly greater than the mean Hausdorff distance for the entire set of trajectories.

The mean Hausdorff distance between each trajectory and its corresponding compressed representation was 108 meters with standard deviation of 82 meters. Given the fact that this works represents the first attempt to compress trajectories using a basis in an unsupervised manner, the authors feel that a mean geometrical accuracy of close to 100 meters is very positive. A close examine of the results revealed that in the majority of cases where the Hausdorff distance was greater than 150 meters this could be attributed to one of the boundaries of the trajectory in question lying in the center of a basis element.

The set of basis elements used for compressed contained 64 elements which in turn contained a total 991 points or 0-simplices. We consider each 0-simplices to be a distinct numerical value. The compression of each trajectory is accomplished by storing a corresponding list of integers in the range [1,64] where a bijection exists between the integers in this range and the basis elements. Considering each integer a numerical value, a total of 8,807 numerical values were required to compress the trajectories. Therefore the total number of numerical values used to represent the basis elements and compress the trajectories was 8,807+9,91=9,798.

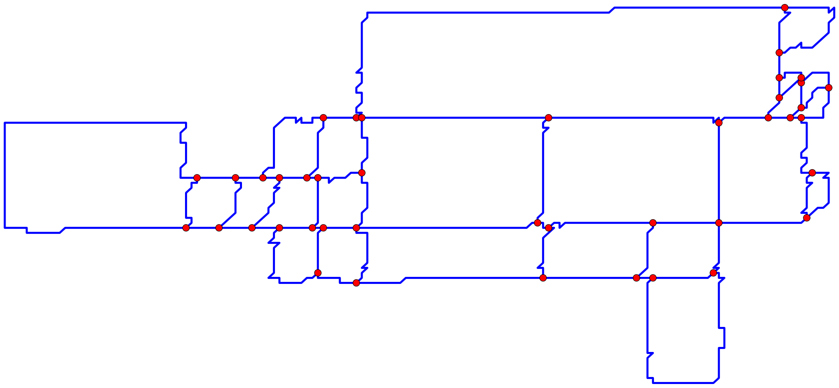
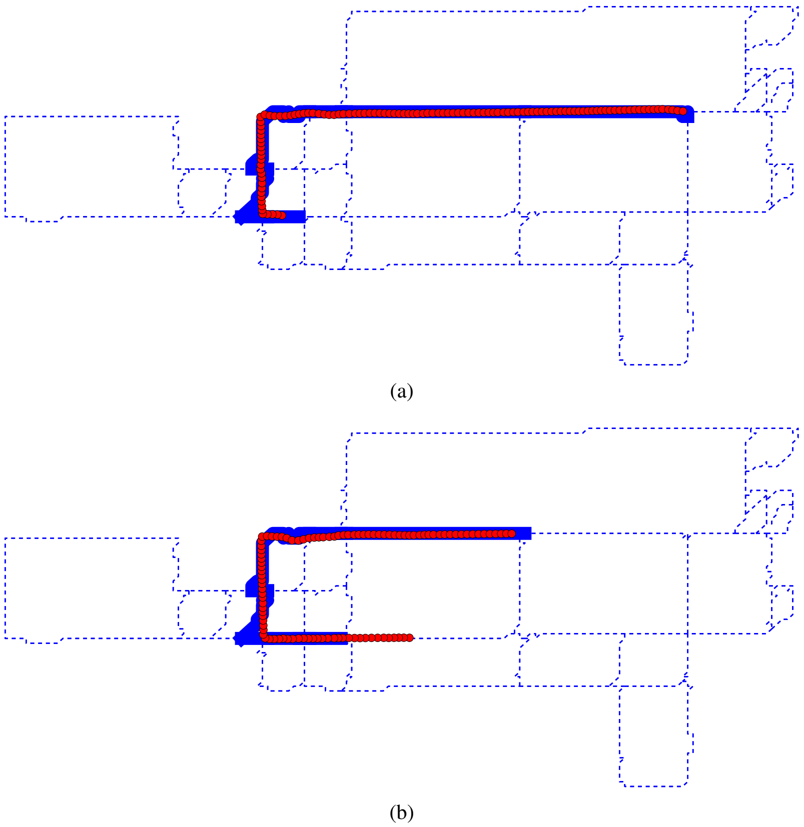


Fig. 3. The set of 0-simplices where multiple paths meets and the set of individual paths between these 0-simplices are represented by red dots and blue lines respectively.



(a)

(b)

Fig. 4. In each of the two above figures the compression of an individual trajectory is illustrated. In both cases the set of GPS points corresponding to the trajectory in question are represented by red dots. The set of basis elements in the corresponding compressed representation are represented by thick solid lines while all other basis elements are represented by thin dashed lines.

On the other hand the original set of trajectories contains 118,221 points and we consider each to be a distinct numerical value. The data compression ratio achieved is therefore approximately 118,221/9,798 = 12.06.

Another good attribute of the proposed compress methodology is its low computational complexity. The steps in the methodology of density estimation, filtration construction, compression basis estimation [34] and compression can all be computed in reasonable time. Our python implementation running on a Laptop containing an Intel Core i7 2.70GHz processor was able to compute the basis for compression and subsequently compress the data used in our evaluation in 96 and 55 seconds respectively.

VIII. CONCLUSIONS

This paper presents a trajectory compression method which robustly determines a suitable basis for compression in an unsupervised manner. The elements in this basis correspond to paths in the space which are topologically distinct. The method was demonstrated in the context of GPS trajectories captured in a city where elements in resulting basis generally correspond to individual streets. The proposed method is distinct from related works which can be considered supervised in nature owing to the fact that they require the basis for compression to be known a priori. The results achieved were very positive with respect to both geometrical accuracy and compression ratio. The proposed method also possess low computational complexity.

In future work the authors hope to evaluate the proposed method in the context of other types of trajectories such as those in a robot configuration space. The proposed method also has potential applications beyond trajectory compression. For example, it is evident from Figure 3 that the method not only determines a suitable basis for compression but also recovers the structure of the underlying transportation network. Steiner et al. [39] demonstrated that knowledge of this underlying network can facilitate map reduction in the context of localization.

References

[1] S. Bhattacharya, M. Likhachev, and V. Kumar, "Topological constraints in search-based robot path planning," *Autonomous Robots*, vol. 33, no. 3, pp. 273–290, 2012.

[2] M. Kuderer, C. Sprunk, H. Kretzschmar, and W. Burgard, "Online generation of homotopically distinct navigation paths," in *IEEE International Conference on Robotics and Automation*, 2014, pp. 6462–6467.

[3] F. T. Pokorny and D. Kragic, "Data-driven topological motion planning with persistent cohomology," in *Robotics: Science and Systems Conference*, 2015.

[4] W. Liu, Y. Zheng, S. Chawla, J. Yuan, and X. Xing, "Discovering spatio-temporal causal interactions in traffic data streams," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 1010–1018.

[5] H. Pham, C. Shahabi, and Y. Liu, "Ebm: an entropy-based model to infer social strength from spatiotemporal data," in *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*. ACM, 2013, pp. 265–276.

[6] M. De Berg, M. Van Kreveld, M. Overmars, and O. C. Schwarzkopf, *Computational geometry*. Springer, 2000.

[7] R. Song, W. Sun, B. Zheng, and Y. Zheng, "Press: A novel framework of trajectory compression in road networks," *Proceedings of the VLDB Endowment*, vol. 7, no. 9, pp. 661–672, 2014.

[8] K. Liu, Y. Li, J. Dai, S. Shang, and K. Zheng, "Compressing large scale urban trajectory data," in *Proceedings of the Fourth International Workshop on Cloud Data and Platforms*, 2014, p. 3.

[9] G. Kellaris, N. Pelekis, and Y. Theodoridis, "Map-matched trajectory compression," *Journal of Systems and Software*, vol. 86, no. 6, pp. 1566–1579, 2013.

[10] F. Schmid, K.-F. Richter, and P. Laube, "Semantic trajectory compression," in *Advances in Spatial and Temporal Databases*. Springer, 2009, pp. 411–416.

[11] R. Ghrist, "Barcodes: the persistent topology of data," *Bulletin of the American Mathematical Society*, vol. 45, no. 1, pp. 61–75, 2008.

[12] H. Edelsbrunner and J. Harer, *Computational topology: an introduction*. American Mathematical Soc., 2010.

[13] A. Zomorodian and G. Carlsson, "Localized homology," *Computational Geometry*, vol. 41, no. 3, pp. 126–148, 2008.

[14] J. M. Phillips, B. Wang, and Y. Zheng, "Geometric inference on kernel density estimates," *arXiv preprint arXiv:1307.7760*, 2013.

[15] F. Chazal, B. T. Fasy, F. Lecci, B. Michel, A. Rinaldo, and L. Wasserman, "Robust topological inference: Distance to a measure and kernel distance," *arXiv preprint arXiv:1412.7197*, 2014.

[16] H. Edelsbrunner, D. Letscher, and A. Zomorodian, "Topological persistence and simplification," *Discrete and Computational Geometry*, vol. 28, no. 4, pp. 511–533, 2002.

[17] P. Corcoran, P. Mooney, and A. Winstanley, "Planar and non-planar topologically consistent vector map simplification," *International Journal of Geographical Information Science*, vol. 25, no. 10, pp. 1659–1680, 2011.

[18] N. Meratnia and A. Rolf, "Spatiotemporal compression techniques for moving point objects," in *Advances in Database Technology-EDBT 2004*. Springer, 2004, pp. 765–782.

[19] Y. Chen, K. Jiang, Y. Zheng, C. Li, and N. Yu, "Trajectory simplification method for location-based social networking services," in *Proceedings of the 2009 International Workshop on Location Based Social Networks*. ACM, 2009, pp. 33–40.

[20] D. Feldman, C. Sung, and D. Rus, "The single pixel gps: learning big data signals from tiny coresets," in *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*. ACM, 2012, pp. 23–32.

[21] O. Salzman, D. Shaharabani, P. K. Agarwal, and D. Halperin, "Sparsification of motion-planning roadmaps by edge contraction," *The International Journal of Robotics Research*, p. 0278364914556517, 2014.

[22] J. Muckell, J.-H. Hwang, C. T. Lawson, and S. Ravi, "Algorithms for compressing gps trajectory data: an empirical evaluation," in *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 2010, pp. 402–405.

[23] C. E. White, D. Bernstein, and A. L. Kornhauser, "Some map matching algorithms for personal navigation assistants," *Transportation Research Part C: Emerging Technologies*, vol. 8, no. 1, pp. 91–108, 2000.

[24] P. Newson and J. Krumm, "Hidden markov map matching through noise and sparseness," in *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems*. ACM, 2009, pp. 336–343.

[25] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk, "On map-matching vehicle tracking data," in *Proceedings of the 31st international conference on Very large data bases*. VLDB Endowment, 2005, pp. 853–864.

[26] B. W. Silverman, *Density estimation for statistics and data analysis*. CRC press, 1986, vol. 26.

[27] F. Chazal, D. Cohen-Steiner, and Q. Mérigot, "Geometric inference for probability measures," *Foundations of Computational Mathematics*, vol. 11, no. 6, pp. 733–751, 2011.

[28] R. Ghrist, *Elementary applied topology*. CreateSpace Independent Publishing Platform, 2014.

[29] A. Hatcher, *Algebraic Topology*. Cambridge University Press, 2002.

[30] J. Lee, *Introduction to topological manifolds*. Springer Science & Business Media, 2010, vol. 940.

[31] F. T. Pokorny, M. Hawasly, and S. Ramamoorthy, "Multiscale topological trajectory classification with persistent homology," in *Robotics: science and systems*, 2014.

[32] A. Zomorodian, "Computational topology," in *Algorithms and theory of computation handbook*. Chapman & Hall/CRC, 2010, pp. 3–3.

[33] S. Kim, K. Sreenath, S. Bhattacharya, and V. Kumar, "Optimal trajectory generation under homology class constraints," in *IEEE Conference on Decision and Control*, 2012, pp. 3157–3164.

[34] C. Chen and M. Kerber, "Persistent homology computation with a twist," in *Proceedings 27th European Workshop on Computational Geometry*. Citeseer, 2011.

[35] A. Zomorodian and G. Carlsson, "Computing persistent homology," *Discrete & Computational Geometry*, vol. 33, no. 2, pp. 249–274, 2005.

[36] H. Edelsbrunner and A. Zomorodian, "Computing linking numbers of a filtration," in *Algorithms in Bioinformatics*. Springer, 2001, pp. 112–127.

[37] H. Edelsbrunner and J. Harer, "Persistent homology - a survey," in *Surveys on Discrete and Computational Geometry. Twenty Years Later*, J. E. Goodman, J. Pach, and R. Pollack, Eds. American Mathematical Society, 2008.

[38] M. Ahmed, S. Karagiorgou, D. Pfoser, and C. Wenk, "A comparison and evaluation of map construction algorithms using vehicle tracking data," *GeoInformatica*, pp. 1–32, 2014.

[39] T. J. Steiner, G. Huang, and J. J. Leonard, "Location utility-based map reduction," in *IEEE International Conference on Robotics and Automation*, 2015, pp. 479–486.