2020 25th International Conference on Pattern Recognition (ICPR)
Milan, Italy, Jan 10-15, 2021

A Framework for Local Outlier Detection from Spatio-Temporal Trajectory Datasets

Xumin Cai*, Berkay Aydin†, Anli Ji‡, Rafal Angryk§

Department of Computer Science, Georgia State University
Email: xcai3@student.gsu.edu*, baydin2@cs.gsu.edu†, aji1@student.gsu.edu‡, rangryk@cs.gsu.edu§

***Abstract*—As one of the primary tasks in data mining, outlier detection serves a significant role in data quality enhancement for the scientific model prediction and revealing the abnormal hidden patterns from large scale trajectory datasets. In this paper, we introduce a versatile framework for detecting local trajectory outliers using spatial and temporal features of moving objects. Our local outlier detection consists of three phases. First, we divide the raw trajectory into trajectory segments by using a time-based partition strategy and extracting trajectory features from spatial attributes for each trajectory segment. Second, we create template trajectory segments based on a clustering schema. Finally, we compute the abnormal score, which measures the dissimilarity among the query and template trajectory segments, and thus determine the outlying trajectory segments according to the overall distribution of the abnormal score. To show the effectiveness of our approach, we conduct two case studies on the real-life solar active region and Coronal Mass Ejection (CME) trajectory datasets. Our results show that our local outlier detection method can successfully detect the reporting errors and anomalous phenomenon in both of our case studies.**

I. Introduction

With the proliferation of mainstream location-based services and surveillance equipment, unprecedented amounts of spatio-temporal trajectory data became available for large-scale analytics tasks. Spatio-temporal trajectory [1] can be defined as the moving objects changing spatial location over time. This complex, often semi-structured, data type has a lot to offer for many pattern recognition tasks in various scientific domains. The presence of outliers makes these pattern recognition tasks challenging as they introduce discordance into the data. In this regard, two main reasons emerge so as to identify outliers: Separating outliers, by improving data accuracy, can improve the performance of predictive modeling and identify rarely occurring, often neglected, data instances can be the main recognition task. In recent years, a variety of outlier detection techniques of spatio-temporal trajectory data emerged and gained increasing interest in various application fields [2]–[4]. Lee et al. proposed a partition and detect framework by using the hybrid of distance-based and density-based approach [2] where the raw trajectory is partitioned by a two-level partition strategy with a minimum description length (MDL) principle. It identifies the outlying sub-trajectories based on their densities. This approach is computationally expensive and may not be suitable for large-scale datasets. Ge et al. [3] proposed an outlier detection method, called TOP-EYE, which continuously calculates the outlying score of the trajectory. This method utilizes the grid-based partition strategy and detects the outlying trajectory by calculating the similarity score between the summarized trajectory and query trajectory. Moreover, Shen et al. [4] detected globally outlying trajectories based on a knowledge-driven approach. This method defines abnormal moving behaviors for vehicle trajectories. Then, it calculates a suspicion score of anomalous events for each trajectory and explores global anomaly trajectories by ranking top-N suspicious events. A recent survey on trajectory outlier analysis techniques is also available in [5].

In this work, we propose a framework that aims to detect local outliers in terms of evolving spatial features of trajectories and segments. This framework has three phases. Firstly, we break up the raw trajectory – by utilizing a temporal partitioning strategy, into several trajectory segments without losing spatial or temporal information from the raw trajectory. For each trajectory segment, we generate summary features (e.g., distance displacement, velocity, acceleration). In the second stage, we cluster the spatio-temporal summary features of trajectory segments and thus generate centroids of each cluster, i.e., the template trajectory segments. In the final stage, we calculate an abnormal score (AB score), which is the weighted sum of the distance between template trajectory segments and query trajectory segments. Finally, we determine

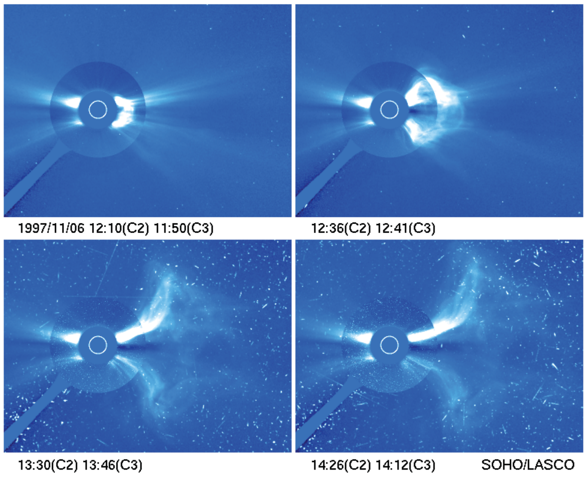


Fig. 1: Evolution of a large CME trajectory. The CME originated from a X9.4 solar flare (30° off the west limb of the Sun) on 6 November 1997. The figures are composite images as seen by the LASCO/SOHO [6]. Courtesy of SOHO/LASCO consortium

978-1-7281-8808-9/20/$31.00 ©2020 IEEE

the outlying trajectory segments, i.e., local outliers, with an empirical threshold based on the AB score distributions.

We also conducted two outlier detection case studies on two real-life datasets from solar astronomy domain to show the effectiveness of our framework. These are solar active region and coronal mass ejection (CME) trajectory datasets. Both of these are critical for space weather forecasting, which can have serious implications for human life [7]. An example CME event and its evolution as a spatio-temporal trajectory is shown in Fig. 1.

The rest of the paper is structured as follows. In Section II, we describe the general framework, partitioning strategy, and clustering approach for detecting the local outliers from the spatio-temporal trajectories. In Section III, we conduct two case studies on real-life datasets from solar physics domain, to demonstrate the effectiveness of our work. In Section IV, we provide our concluding remarks and discuss future work.

II. General Framework

A spatio-temporal trajectory is defined as a sequence of chronologically ordered time-geometry pairs and denoted by $TR = \{\langle t_1, g_1\rangle, \langle t_2, g_2\rangle, \ldots, \langle t_j, g_j\rangle\}$ where $t_1 < t_2 < \cdots < t_j$, and t_j represents timestamp or time interval, g_j represent d-dimensional spatial objects [8]. Formally, the outlying trajectory [9] is defined as the trajectory or sub-trajectory which is significantly different from the majority trajectories in the dataset. In our work, the local outlier in trajectory is the outlying trajectory segments with the descriptive summary features significantly deviating from the other trajectory segments (ts). In Fig. 2, a schematic diagram illustrates the workflow of our local outlier detection framework. Firstly, each TR object is represented as the k trajectory segments $\{ts_1, ts_2, \ldots, ts_k\}$ and segments are converted to predetermined summary features. Secondly, we generate template summary features of ts by applying a clustering schema to the summary features of trajectory segments. In the final phase, the outlying trajectory segments, denoted as ots, is determined by the overall distribution of abnormal scores. We will discuss each phase step by step in the following subsections.

A. Temporal Partitioning Strategy and Feature Extraction

It is ideal to partition the raw trajectory which has consistent, periodic sampling interval into k trajectory segments with an equi-length time interval. However, in real-life applications, the sampling interval can be inconsistent and non-periodic due to the limitations of recording equipment or the preferences of data collectors. Both periodic and non-periodic sampling interval scenarios need to be taken into account when implementing the partition strategy. Algorithm 1 shows the partition algorithm in our framework. For the TR with periodic sampling, we first apply linear interpolation for the trajectory whose sampling interval if there are any missing values. We consider the sampling interval ΔT as the basic time bin width and use $\Delta T * n$ as the partition time bin width. We also apply linear spatial interpolation to estimate the approximate spatial data for a trajectory whose time interval is not consistent. Fig. 3 shows an example partition process for a periodically sampled trajectory (sampling interval is ΔT). Note here that partitioning schema accounts for missing geometry records, which is often the case for trajectory datasets, using a spatio-temporal interpolation procedure (see g_6, g_7, and g_8 are not recorded in Fig. 3).

Algorithm 1 Temporal partitioning algorithm

```
Input: Trajectory – $TR = \{\langle t_1, g_1\rangle, \ldots, \langle t_j, g_j\rangle\}$
    $k$, $n$, $minp$
Output: Trajectory as a set of trajectory segments –
    $TR = \{ts_1, ts_2, \ldots, ts_k\}$
if $TR$ has periodic sampling then
    if $\frac{t_j.end - t_j.start}{\Delta T} \neq j$ then
        Estimate $TR$ spatial location by using
          linear interpolation
    end if
    $time\_bin\_width = \Delta T * n$
    Partition $TR$ at $time\_bin\_width$
    return $TR = \{ts_1, ts_2, \ldots, ts_k\}$
else if $TR$ has non-periodic sampling then
    for $i = k$ to $1$ do
        $time\_bin\_width = \frac{t_j.end - t_j.start}{k}$
        Partition $TR$ at $time\_bin\_width$
        if $\min(number\ of\ \langle t_j, g_j\rangle$ in each
          time bin$) \geqslant minp$ then
            return $TR = \{ts_1, ts_2, \ldots, ts_k\}$
        end if
    end for
end if
```

For the TR with non-periodic sampling intervals, we use the lifespan of trajectory divided into k segments to find a near-optimal time-bin width and ensure that there exists sufficient number of time-geometry pairs in the segment. The minimum number of time-geometry pairs is denoted as $minp$ and is given as parameter to temporal partitioning procedure. Fig. 4 shows an example partition process illustration for an arbitrary trajectory sampled at nonuniform time intervals.

Thus, each TR in the dataset is partitioned into a collection of consecutive trajectory segments and denoted by $TR = \{ts_1, ts_2, ..., ts_k\}$ and each segment, ts_k, contains multiple time-geometry pairs, $\langle t_j, g_j\rangle$, denoted by $ts_k = \langle t_m, g_m\rangle, \ldots, \langle t_n, g_n >$ where $t_m < t_{m+1} < t_n$. For each trajectory segment, ts_k, we extract application-dependent descriptive features. These are spatial features that reflect the spatio-temporal characteristics of the segment during the time period from t_m to t_n. These can include, but are not limited to total distance covered, average velocity, or average acceleration. The vectors of spatial features representing the characteristic of ts_k will be used for clustering segments in the next phase. These features do not necessarily have to be spatial and can be any time-dependent feature, but for the context of local trajectory outlier detection, we consider only spatial features.

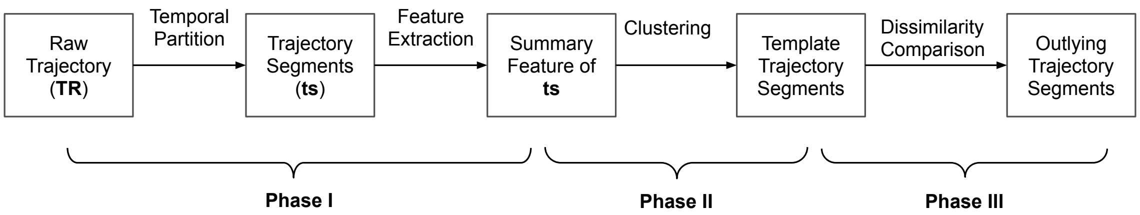


Fig. 2: Overall workflow of the local trajectory outlier detection framework. Our method starts with partitioning and feature extraction, then determines the outliers based on cluster centers serving as templates. The segments are ranked using the abnormal score, which effectively checks the dissimilarity.

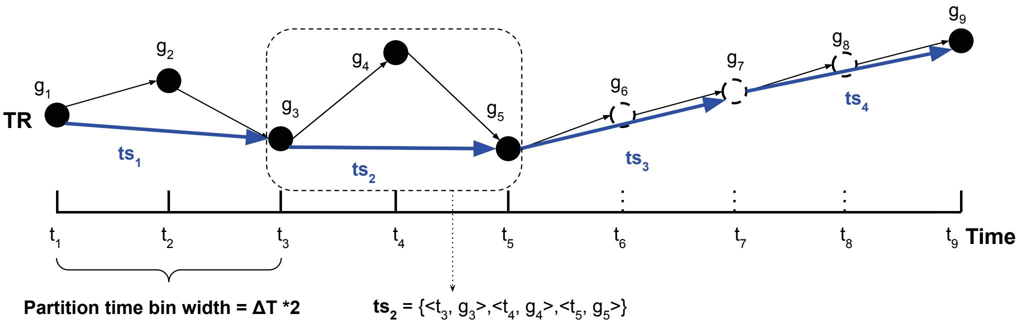

TR
g1
g2
g3
g4
g5
g6
g7
g8
g9
ts1
ts2
ts3
ts4
t1
t2
t3
t4
t5
t6
t7
t8
t9
Time
Partition time bin width = ΔT *2
ts2 = {<t3, g3>,<t4, g4>,<t5, g5>}

Fig. 3: A $TR = \langle t_1, g_1\rangle, \langle t_2, g_2\rangle, ..., \langle t_9, g_9\rangle$ with periodic sampling interval T, (1) we estimate the approximate spatial location (the dash-dot) during $[t_5, t_9]$ by using linear interpolation and (2) partition the TR at $\Delta T * 2$, thus $TR = ts_1, ts_2, ..., ts_4$ and each ts_k contains three $\langle t_j, g_j\rangle$ pairs.

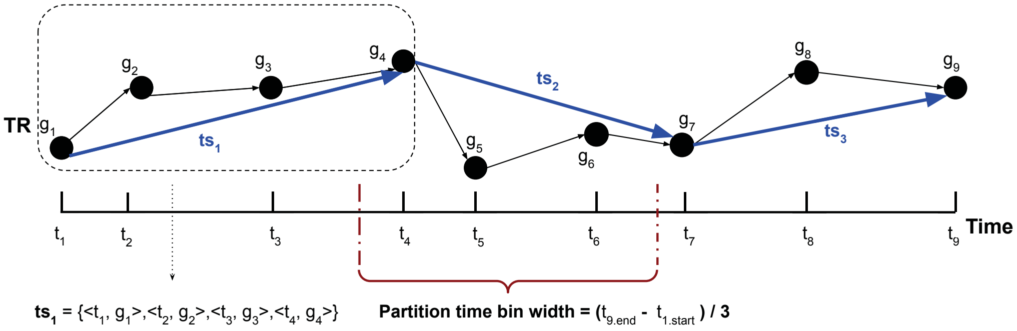

TR
g1
g2
g3
g4
g5
g6
g7
g8
g9
ts1
ts2
ts3
t1
t2
t3
t4
t5
t6
t7
t8
t9
Time
ts1 = {<t1, g1>,<t2, g2>,<t3, g3>,<t4, g4>}
Partition time bin width = (t9.end - t1.start) / 3

Fig. 4: A $TR = \langle t_1, g_1\rangle, \langle t_2, g_2\rangle, ..., \langle t_9, g_9\rangle$ with a nonuniform time interval, (1) we set the minimum number of $\langle t_j, g_j\rangle$ pairs in each time bin is $minp = 3$ and get $\frac{t_9.end - t_1.start}{3}$ as the partition time bin width during$[t_1, t_9]$ based on the partition algorithm. (2) partition the TR at $\frac{t_9.end - t_1.start}{3}$, thus $TR = \{ts_1, ts_2, ts_3\}$ and each ts_k contains minimum three $\langle t_j, g_j\rangle$ pairs.

B. Trajectory Segments Clustering and Template Generation

The goal in this phase is to generate template trajectory segments by applying a clustering algorithm to the extracted features from segments. For simplicity, we will use the centroid-based K-means++ clustering algorithm; however, any clustering algorithm can be used as part of the framework. K-means [10] is a widely used unsupervised learning model that aims to divide the dataset into K non-overlapping groups. It assigns the observation to the nearest centroid of clusters based on a distance measure and minimizes the inter-cluster variance. Each cluster centroid is the mean of observations in each cluster. K-means++ [11] is an extension of K-means with an improved centroid initialization strategy. K-means++ initializes the first centroid from the dataset, and selects the remaining centroids by calculating the probabilities with respect to the squared distances from the existing centroid or centroids. Another consideration for more robust clustering in K-means++ algorithm is finding a suitable K. While the number of clusters should not significantly impact our overall outlier detection procedure, to obtain the optimal K

for the clusters, we use the empirical Elbow method [12] and Silhouette Analysis [13] as the evaluation method. The Elbow method measures the inter-cluster variance namely inertia by calculating Sum Square Error (SSE) [12] for each candidate value K. The goal is to empirically find the K value which substantially reduces SSE from $K-1$, but does not significantly improve for $K+1$. Silhouette Analysis [13] is another method to evaluate cluster quality. It measures the cohesion within the cluster and separation outside the cluster. The optimal K is obtained by selecting the K resulting in the maximum Silhouette coefficient [13]. We apply the K-means++ clustering to the summary feature vectors of overall trajectory segments extracted from the first phase. Each cluster will ideally represent the trajectory segments with similar movement characteristics and the centroid of each cluster reflects the mean feature vectors of the corresponding trajectory segments. We will use the centroid as the template feature, which essentially is the prototype trajectory segment.

C. Dissimilarity Comparison

To quantify the similarity among the template and query trajectory segments, we introduce *the abnormal score* (AB score) which is the weighted sum of the Euclidean distance between the query trajectory segments ts_i and each template trajectory segment, i.e., the centroid c_j.

$$AB_i = \sum_{j=1}^{K} w_j * dist(ts_i, c_j) \tag{1}$$

Where K is the number of clusters we select from the second phase, and w_j is calculated as the ratio between the number of trajectory segments in each cluster and the total number of trajectory segments in the datasets.

$$w_j = \frac{\textit{Number of trajectory segments in } c_j}{\textit{Total number of trajectory segments}} \tag{2}$$

The AB score indicates how far the ts_i is to the set of template trajectory segments. The templates essentially show the movement trends for each cluster using the summary features. The lower AB score shows that the query ts_i is closer to sufficiently large number of the trajectory segments. While the higher AB score shows that the summary features of the query largely deviates from the majority of the segments in the dataset. In the case of significantly high AB scores, ts_i is more likely to be an outlying trajectory segment, which requires us to set a threshold.

$$ots = \begin{cases} \textit{True} & \textbf{if } AB_i \geqslant threshold \\ \textit{False} & \textbf{if } AB_i < threshold \end{cases} \tag{3}$$

We determine this AB score *threshold* of outlying trajectory segments again empirically as it is mostly domain-dependent. This is done by analyzing the distribution of the AB score in this study but it can also be done by individually checking borderline cases. Finally, the ts_i with the AB score above the threshold is marked as the local outlier.

TABLE I: Spatial Features used in two experiments

No.	Spatial Feature	Formula
1	Lon. Displacement	$ts_i.x_{end} - ts_i.x_{start}$
2	Lat. Displacement	$ts_i.y_{end} - ts_i.y_{start}$
3	Displacement Vector Magnitude	$\lVert \vec{ts_i} \rVert$
4	Displacement Vector Direction	$tan^{-1}(\frac{ts_i.y_{end} - ts_i.y_{start}}{ts_i.x_{end} - ts_i.x_{start}})$
5	Average Velocity (height)	$\frac{ts_i.height_{end} - ts_i.height_{start}}{ts_i.time_{end} - ts_i.time_{start}}$
6	Average Acceleration (height)	$\frac{d^2(g_{m+1}.height - g_m.height)}{d(t_{m+1} - t_m)^2}$
7	Time-normalized Cumulative Angle Displacement	$\frac{\sum_{m=1}^{n} \min(\Delta\alpha\%360^\circ, -\Delta\alpha\%360^\circ)}{ts_i.time_{end} - ts_i.time_{start}}$, where $\Delta\alpha = g_{m+1}.angle - g_m.angle$

III. Evaluation

In this section, we conduct two case studies on two real-life datasets from solar astronomy domain: (1) the solar active region trajectory dataset from National Oceanic and Atmospheric Administration (NOAA), and (2) Coronal Mass Ejection (CME) events trajectory dataset from NASA / Goddard Space Flight Center. Our case studies are performed primarily to demonstrate the effectiveness of our local outlier detection framework.

We retrieved the solar active region trajectory dataset from [14]. In this dataset, heliographic longitudes and latitudes of the solar active region centroids are reported daily along with additional non-spatial metadata. The solar active regions are collected between January 1996 to August 2019. There are 4,795 trajectories with at least two daily observations and a total of 45,319 time-geometry pairs.

We obtained the CME dataset from [15] between January 1996 to March 2019, and use the height and angle as spatial attributes in sky-plane coordinate system. The angle is the position angle (in degrees) with respect to Sun's center from observer's field of view, while the height represents the distance between the Sun's corona and the CME in R_{Sun}, which is the radius of the Sun (approx. 695,700 kms). We disregarded the faint CMEs with less than ten records. In the end, we have 16,509 CME trajectories and 372,048 time-geometry pairs records in this experiment. CME locations are recorded in non-periodic time intervals, time cadence ranging from tens of seconds to several hours.

A. Solar Active Region Trajectory

The time-geometry pairs of solar active regions are reported daily ($\Delta T = 24$ hours). Due to this relatively low-frequency in reporting, we set n=1 as the input parameter in the temporal partition algorithm (meaning only one time interval with start and end geometries will constitute a segment). Each trajectory is partitioned into multiple ts and each ts contains two time-geometry pairs. In the end, we have 40,758 trajectory segments after initial preprocessing, interpolation, and temporal partition. For each ts, we generate four normalized vector-based spatial features, namely, longitudinal displacement, latitudinal displacement, displacement vector magnitude, and displacement vector direction, shown in Table I (features 1 through 4). We chose $K = 3$ as the number of the clusters based on the elbow method and mean Silhouette score shown in Fig. 5. Based on the given features and the empirical K value, we clustered the trajectory segments into three clusters. The summary statistics for each cluster is shown in Table II. A strong majority (~99.5%) of the solar active region trajectory segments are clustered into Cluster 0, which has an average longitudinal displacement of $+13.33^\circ$ and these segments barely change their latitudes and vector direction. This is the expected movement of solar active regions, caused by the solar rotation (covering 180° in 13-14 days). The segments in Cluster 1 and Cluster 2 represent the minority (both totalling ~0.5%), whose spatio-temporal features are vastly different from the ones in Cluster 0 (e.g. dramatic changes (over $\pm 100^\circ$) in vector direction). Note here that while Cluster 1 and 2 represent rather anomalous movement behaviors, the aim of the clustering step is not to find outlier clusters, and outlying trajectory segments are found by the next step using the AB score.

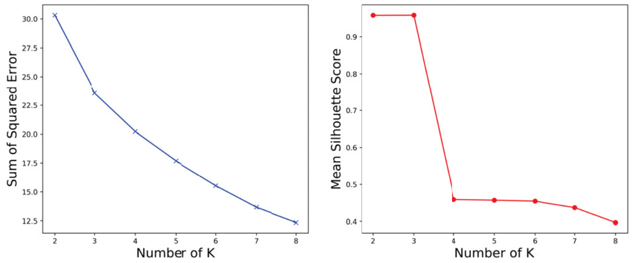


Fig. 5: Sum of squared errors for the Elbow method (on the left) and Mean Silhouette Score (on the right) for determining number of clusters (K) for solar active region trajectory segments. We selected $K = 3$.

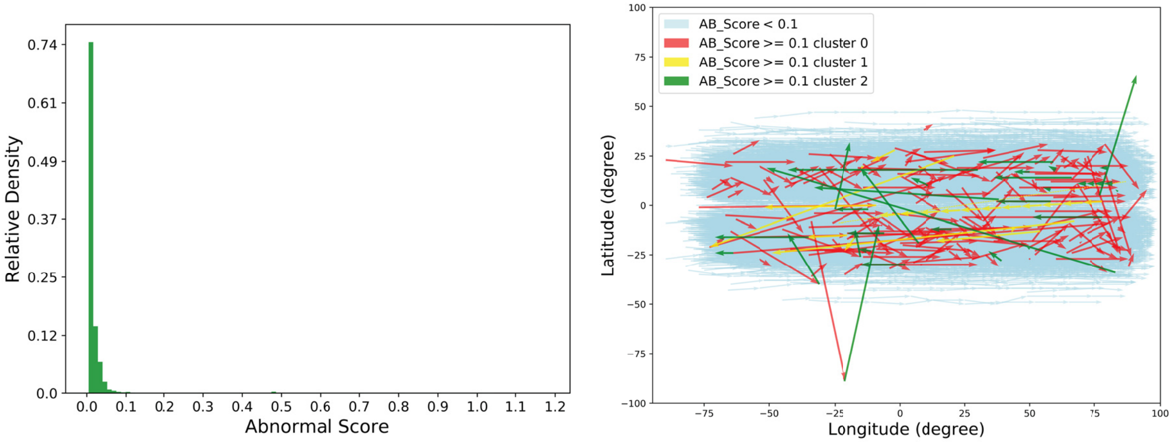
TABLE II: The summary statistics for three solar active region trajectory segment clusters. The count is the number of trajectory segments in each cluster, mean and std is the average value and the standard deviation of four spatial features in the cluster.

Cluster 0 `count=40,586`	Lon. displacement (deg)	Lat. displacement (deg)	Disp. Vector Magnitude	Disp. Vector Direction
mean	13.33	0.01	13.38	0.06
std	1.95	1.22	2.01	4.75
Cluster 1 `count=140`	Lon. displacement (deg)	Lat. displacement (deg)	Disp. Vector Magnitude	Disp. Vector Direction
mean	-3.72	-0.82	3.89	-168.49
std	13.39	4.16	14.00	8.56
Cluster 2 `count=32`	Lon. displacement (deg)	Lat. displacement (deg)	Disp. Vector Magnitude	Disp. Vector Direction
mean	-16.59	10.19	25.22	156.13
std	28.81	20.48	31.38	34.00

In the dissimilarity comparison phase, we use the AB score discussed in II-C and obtain the abnormal score distribution for the trajectory segments, shown in Fig. 6a. We find that over 99% of the AB scores are below 0.1, hence, in this case, we set the threshold as 0.1 and get 354 outlying trajectory segments namely, the local outliers. In Fig. 6b, the light blue movement vectors (in the background) represent the normal ts, while red, yellow, and green vectors represent the ots that come from corresponding clusters (0, 1, and 2 respectively). We can see that the magnitudes of normal ts, which are essentially uniform and move from the east to the west-limb (east-west direction is reversed for solar coordinates) with slight direction changes (generally $\leqslant \pm 2^\circ$). Among the ots, we can see that the majority moving direction of ots from Cluster 0 is the same with normal ts, but with anomalous magnitudes, and the ones from Cluster 1 (yellow) and Cluster 2 (green) shows the anomalous behavior in both moving direction and magnitude; i.e., the opposite direction to solar rotation and unexpected lengths compared to the normal ts. In our previous work [16], we showed that there are around 60 anomalous NOAA active region trajectories (global) between 2010 and 2018, which are caused by the erroneous location reporting. The detected outliers in this case study include all of the previous reporting errors, which verifies the reliability of our outlier detection methodology.

B. Coronal Mass Ejection (CME) trajectory

Our second case study is on coronal mass ejections (CMEs). The sampling interval (time cadence) of CME trajectories is non-uniform and vastly irregular (from seconds to several hours mostly due to the cadence of LASCO instrument onboard SOHO spacecraft [17]). To this end, we use the temporal partition algorithm for non-periodic sampling. In this case, we are interested in three spatial features: average velocity, average acceleration, and cumulative angle displacement, described in Table I (features 5 to 7). Hence, in the partitioning phase, we set input parameters $k = 15$ and $minp = 3$ to ensure the minimum number of time-geometry records in each ts is three. After trajectory segmentation, we generate 55,976 trajectory segments with three summary features (features are then normalized). Based on the elbow method and Silhouette Analysis, we choose $K = 3$ for K-means clustering (scores shown in Fig. 7) and create three clusters. About $\sim 72\%$ of segments belong to Cluster 0, while $\sim 23\%$ and $\sim 5\%$ of them belong to Clusters 1 and 2, respectively. In this case, based on the distribution of AB scores shown in Fig. 8a, we select top-1% AB score as the threshold for outlying segments (ots). Fig. 8b shows the distribution of three summary features of normal ts and ots from each cluster. It is worth to notice that compared to the summary features of normal ts, the red ots from Cluster 0 shows very slow CMEs (low-velocity), the orange ones from Cluster 1 represents very fast CMEs, and the purple ots with large angle change is from Cluster 2. To better illustrate the movement characteristics among the normal ts and ots in CME datasets, we create the height-angle plots of ts on the polar coordinate plane and corresponding summary statistics are shown in Fig. 9. We randomly choose 0.1% percent of normal ts for improving the visibility and



(a)

(b)

Fig. 6: (a) The distribution of AB scores for solar active region trajectory segments and (b) 2D scatter plot of movement vectors (each showing daily movement) for normal ts (in blue) and outliers (in red, yellow and green).

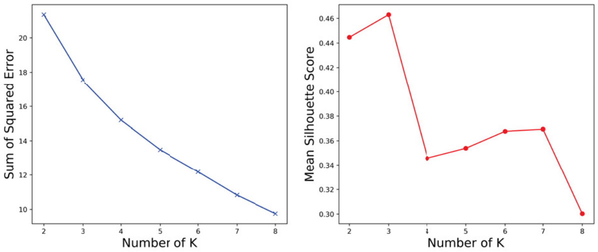


Fig. 7: Sum of squared errors for the Elbow method (on the left) and Mean Silhouette Score (on the right) for determining number of clusters (K) for CME trajectory segments. We selected $K = 3$.

10% of ots from each cluster to demonstrate their outlying spatio-temporal characteristics. We can see that, ots in the Fig. 9b represents the slower CME segments compared to normal ts in Fig. 9a. Similarly, the faster ts is identified as the ots in the Cluster 1 in Fig. 9c. In addition, ts in Fig. 9d shows the zigzag movement patterns which indicates an anomalous moving behavior for a CME (or a reporting error) of ts compared to normal ts.

IV. Concluding Remarks and Future Work

In this paper, we proposed a framework for detecting local outliers from the spatio-temporal trajectory datasets. We introduced a temporal partition algorithm for the trajectories with both periodic and non-periodic recording intervals. By doing that, we aimed to keep the spatial and temporal data intact and to extract the summary features for trajectory segments used in the clustering phase. The template trajectory segments generated by clustering schema indicate the representative moving characteristics in the overall trajectory segments. We also introduced the AB score, which can robustly quantify the dissimilarity between the majority and/or minority moving characteristics of the trajectory segments.

Our outlier detection case studies on solar active region and CME trajectory datasets demonstrate that we can successfully identify the local outliers in these real-life dataset using a simple threshold for AB score. Detailed local outlier detection results show these outliers are most likely the results of either a reporting error or an anomalous movement in these trajectories. Solar active regions and CMEs are essential input data for space weather prediction and modeling. We believe that quality assessments of these event reports, which are often taken for granted, can seriously impact the large-scale solar flare or eruption prediction models.

In the future work, we plan to analyze the impact of different clustering algorithms and distance metrics and extend our work to network-based trajectory datasets.

V. Acknowledgments

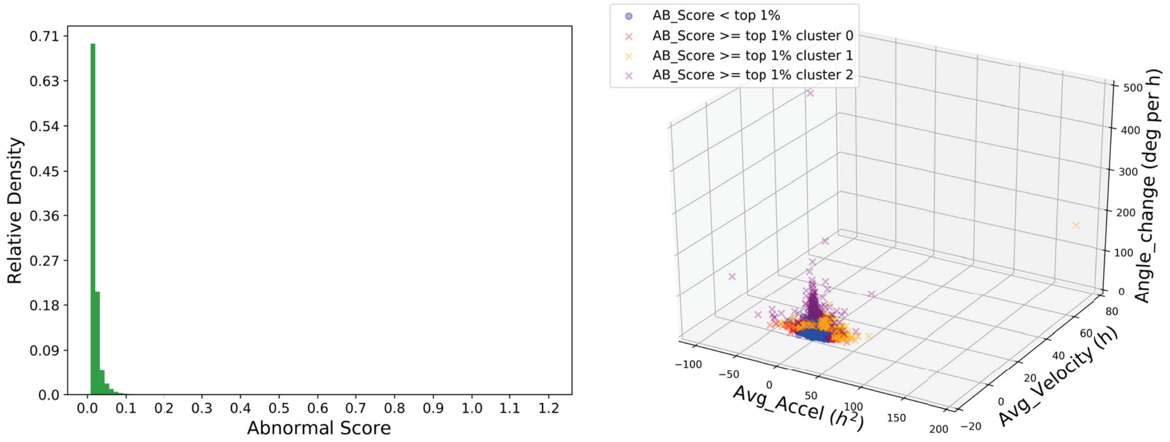
This project has been supported in part by funding from the Division of Advanced Cyberinfrastructure within the Directorate for Computer and Information Science and Engineering, under NSF award #1931555. It was also supported in part by funding from the Heliophysics Living With a Star Science Program, under NASA award #NNX15AF39G.

References

[1] R. H. Güting, M. H. Böhlen, M. Erwig, C. S. Jensen, N. A. Lorentzos, M. Schneider, and M. Vazirgiannis, "A foundation for representing and querying moving objects," *ACM Trans. Database Syst.*, vol. 25, no. 1, p. 1–42, Mar. 2000. [Online]. Available: https://doi.org/10.1145/352958.352963

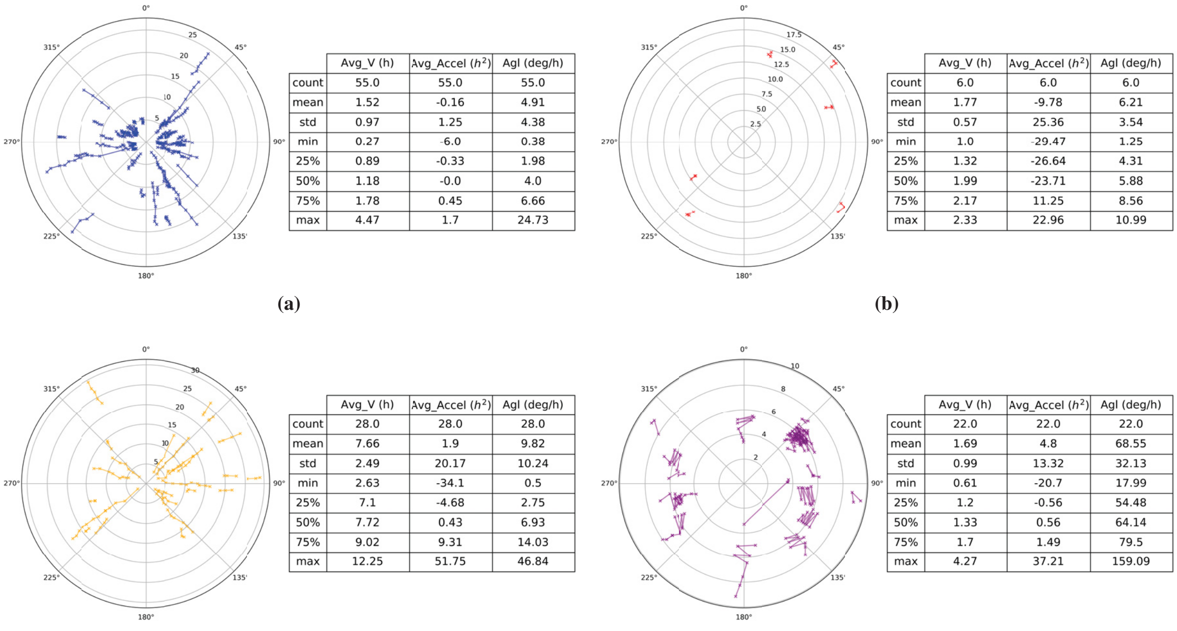
[2] J. Lee, J. Han, and X. Li, "Trajectory outlier detection: A partition-and-detect framework," in *2008 IEEE 24th International Conference on Data Engineering*, 2008, pp. 140–149.

[3] Y. Ge, H. Xiong, Z.-h. Zhou, H. Ozdemir, J. Yu, and K. C. Lee, "Top-eye: Top-k evolving trajectory outlier detection," in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, ser. CIKM '10. New York, NY, USA: Association for Computing Machinery, 2010, p. 1733–1736. [Online]. Available: https://doi.org/10.1145/1871437.1871716



(a) **(b)**

Fig. 8: (a) The distribution of AB scores for CME trajectory segments and (b) 3D scatter of spatial features of normal ts (in blue) and ots instances (in purple, orange and red) from each cluster.



	Avg_V (h)	Avg_Accel (h^2)	Agl (deg/h)
count	55.0	55.0	55.0
mean	1.52	-0.16	4.91
std	0.97	1.25	4.38
min	0.27	-6.0	0.38
25%	0.89	-0.33	1.98
50%	1.18	-0.0	4.0
75%	1.78	0.45	6.66
max	4.47	1.7	24.73

(a)

	Avg_V (h)	Avg_Accel (h^2)	Agl (deg/h)
count	6.0	6.0	6.0
mean	1.77	-9.78	6.21
std	0.57	25.36	3.54
min	1.0	-29.47	1.25
25%	1.32	-26.64	4.31
50%	1.99	-23.71	5.88
75%	2.17	11.25	8.56
max	2.33	22.96	10.99

(b)

	Avg_V (h)	Avg_Accel (h^2)	Agl (deg/h)
count	28.0	28.0	28.0
mean	7.66	1.9	9.82
std	2.49	20.17	10.24
min	2.63	-34.1	0.5
25%	7.1	-4.68	2.75
50%	7.72	0.43	6.93
75%	9.02	9.31	14.03
max	12.25	51.75	46.84

(c)

	Avg_V (h)	Avg_Accel (h^2)	Agl (deg/h)
count	22.0	22.0	22.0
mean	1.69	4.8	68.55
std	0.99	13.32	32.13
min	0.61	-20.7	17.99
25%	1.2	-0.56	54.48
50%	1.33	0.56	64.14
75%	1.7	1.49	79.5
max	4.27	37.21	159.09

(d)

Fig. 9: The movement characteristics of (a) 55 normal ts, (b) 6 outlying trajectory segments (ots) of Cluster 0 , (c) 28 ots of Cluster 1, (d) 22 ots of Cluster 2.

[4] M. Shen, D.-R. Liu, and S.-H. Shann, "Outlier detection from vehicle trajectories to discover roaming events," *Inf. Sci.*, vol. 294, pp. 242–254, 2015.

[5] F. Meng, G. Yuan, S. Lv, Z. Wang, and S. Xia, "An overview on trajectory outlier detection," *Artificial Intelligence Review*, 02 2018.

[6] "Soho-gallery: Best of soho," https://sohowww.nascom.nasa.gov/gallery/SolarCorona/las016.html, (Accessed on 04/15/2020).

[7] M. Moldwin, *An Introduction to Space Weather*. Cambridge University Press, 2008. [Online]. Available: https://doi.org/10.1017/cbo9780511801365

[8] B. Aydin and R. A. Angryk, *Modeling Spatiotemporal Trajectories*. Cham: Springer International Publishing, 2018, pp. 9–15. [Online]. Available: https://doi.org/10.1007/978-3-319-99873-2_2

[9] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.

[10] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. USA: Prentice-Hall, Inc., 1988.

[11] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '07. USA: Society for Industrial and Applied Mathematics, 2007, p. 1027–1035.

[12] D. J. Ketchen and C. L. Shook, “The application of cluster analysis in strategic management research: An analysis and critique,” *Strategic Management Journal*, vol. 17, no. 6, pp. 441–458, 1996. [Online]. Available: http://www.jstor.org/stable/2486927

[13] P. J. Rousseeuw, “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis,” *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53 – 65, 1987. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0377042787901257

[14] NOAA/National Weather Service, ftp://ftp.swpc.noaa.gov/pub/warehouse/, (Accessed on 04/14/2020).

[15] CDAW Data Center, “Soho lasco cme catalog,” https://cdaw.gsfc.nasa.gov/CME_list/, (Accessed on 04/14/2020).

[16] X. Cai, B. Aydin, M. K. Georgoulis, and R. Angryk, “An application of spatio-temporal co-occurrence analyses for integrating solar active region data from multiple reporting modules,” in *2019 IEEE International Conference on Big Data (Big Data)*, 2019, pp. 4950–4959.

[17] N. Gopalswamy, S. Yashiro, G. Michalek, G. Stenborg, A. Vourlidas, S. Freeland, and R. Howard, “The SOHO/LASCO CME Catalog,” *Earth Moon and Planets*, vol. 104, no. 1-4, pp. 295–313, Apr. 2009.