Semantic Segmentation of AIS Trajectories for Detecting Complete Fishing Activities

Song Wu
Université Libre de Bruxelles
Brussels, Belgium
song.wu@ulb.be

Esteban Zimányi
Université Libre de Bruxelles
Brussels, Belgium
esteban.zimanyi@ulb.be

Mahmoud Sakr
Université Libre de Bruxelles
Brussels, Belgium
mahmoud.sakr@ulb.be

Kristian Torp
Aalborg University
Aalborg, Denmark
torp@cs.aau.dk

2022 23rd IEEE International Conference on Mobile Data Management (MDM) | 978-1-6654-5176-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/MDM55031.2022.00092

***Abstract*—Detection of fishing activities in trajectory data is important for authorities to develop fishery management policies and combat illegal, unreported, and unregulated (IUU) fishing at sea. However, the complex movement patterns of fishing activities challenge existing trajectory segmentation approaches, which may not identify complete fishing activities. In light of this, we propose a window-based trajectory segmentation algorithm which aims to detect fishing activities as completely as possible. Firstly, we introduce a visualization-based technique *TPoSTE* to help design features characterizing different movement patterns. Secondly, a window-based segmentation algorithm *WBS-RLE* is proposed to split a trajectory into fishing and non-fishing segments. *WBS-RLE* first utilizes a pre-trained classifier to label windows in a trajectory as fishing or non-fishing, then it uses the run-length encoding technique to merge those labeled windows into complete fishing activities. The effectiveness of our approach and its advantages over existing approaches are evaluated on a real-world trajectory dataset.**

I. Introduction

The sustainable utilisation of ocean resources has been challenged by illegal, unreported and unregulated (IUU) fishing in recent years [1]. IUU fishing not only causes over-exploitation of fishing stocks but also damages the marine environment. Given this, governments like the EU have set up regulations[1] to track and fight IUU fishing. To enforce these regulations, an important step is to identify when and where ships have conducted fishing activities. And this step can be naturally modeled as a trajectory segmentation problem to split ship trajectories into fishing and non-fishing segments.

Although many trajectory segmentation algorithms have been proposed in the literature [2], [3], [4], [5], [6], some limitations exist when they are applied to detect fishing efforts.

Firstly, some of them [4], [5] view trajectories as moving objects travelling among different places, and thus aim to split trajectories into a series of stops and moves. However, fishing activities at sea can exhibit complex movement patterns and may not be simply treated as stops or moves. For example, the complete fishing activity in Fig. 1 was composed of three petal-like parts with alternating high and low speeds, and between every two consecutive parts the ship stopped for some time. So one of our goals in this work is to detect as complete fishing activities as possible.

[1]https://ec.europa.eu/oceans-and-fisheries/fisheries/rules/illegal-fishing_en

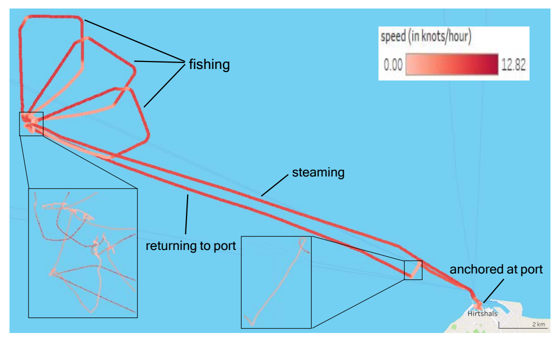

Fig. 1. A typical trajectory by a fishing ship in three stages: steaming to the fishing ground, fishing, and returning to the port

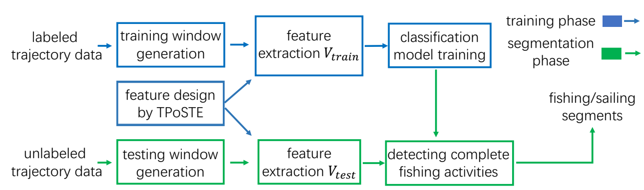

Fig. 2. Overall workflow of the proposed methodology

Secondly, most existing work [2], [3], [6], [7] does not provide labeling annotations. So these approaches can not tell immediately if there are any fishing activities in corresponding trajectories. In light of this, we resort to a supervised approach in this work so that returned trajectory segments contain explicit labels regarding if fishing activities have happened.

Thirdly, one common assumption in the literature (e.g., [8], [2]) is that the resulting segments should have high homogeneity w.r.t. some spatiotemporal criteria or features of trajectory points within segments. For example, two features including speed and direction variation are used in [2] to find suitable partitioning positions in the trajectory. However, ships may move at both high and low speeds during fishing as shown in Fig. 1, and moving at low speeds does not necessarily mean ongoing fishing activities (the right box in Fig. 1). In addition, the movement patterns of fishing efforts at sea may differ greatly depending on the gear type used [1] and the situation on the spot, hence it is non-trivial to design effective criteria to capture the variety of fishing movement patterns.

2375-0324/22/$31.00 ©2022 IEEE
DOI 10.1109/MDM55031.2022.00092

With the above considerations, the goal of this work is to provide a general approach to detect complete fishing activities in ship trajectories. To this end, our main contributions are:

- a technique based on visualization to design features characterizing different movement patterns.
- a semantic segmentation method *WBS-RLE* that can split a ship's trajectory into fishing and non-fishing segments.
- the effectiveness of our approach is evaluated on a real world trajectory dataset.

II. Related Work

There are mainly two lines of research closely related to our work. The first line of research focuses on trajectory points and try to assign each trajectory point a label of fishing or non-fishing [9], [10]. The second line of research [6], [7], [2], [3], [11], [4], [5] deals with the trajectory segmentation problem and aims to partition trajectories into a series of segments according to some spatiotemporal criteria or a cost function.

In the first line of research, speed has been used as the main criterion to recognize fishing activities. Such methods detect fishing activities by using fixed speed ranges [10], or by modeling the speed range as a bi-modal distribution [9]. However this assumption mainly works for trawlers and does not apply to other gear types. Moreover, Fig. 1 highlights that speed alone may not be sufficient to decide if a ship is conducting fishing activities. So this line of research is more suitable for deriving the overall spatiotemporal distribution of fishing efforts as in [9], [10], but not appropriate for the detection of complete fishing activities.

The second line of research falls into two broad categories: (A) studies in [4], [5], [11] follow the conceptual model of trajectories in [12] and aim to split a trajectory into a sequence of stops and moves; whereas (B) other work in [6], [7], [3], [2] does not follow this model and splits trajectories based on some spatiotemporal criteria or homogeneity within segments.

For example, CB-SMoT [4] uses speed as the main criterion to detect stops, and it modifies DBSCAN [13] in two aspects: (1) a point's neighborhood is defined along the trajectory; (2) a point is considered a core point only when the passing time within its neighborhood is above a specified time threshold. However, CB-SMoT is not applicable to our problem because a low speed does not necessarily represent ongoing fishing activities. In comparison, DB-SMoT [5] uses direction change as the main criterion to detect fishing stops. However, DB-SMoT ignores the fact that a complete fishing activity may also include points with little direction change (Fig. 1). For the same reason, the method in [11], which is a combination of CB-SMoT and DB-SMoT, does not apply to our problem.

Among approaches that do not follow the model of stops and moves, Warped K-Means [6] is a variant of the K-Means algorithm to cope with the sequential nature of trajectory data and its main characteristic is that only two clusters are checked in each step. The main limitation of Warped K-Means is that it also requires the number of segments to be given as input, which is usually unknown in practice. SWS [7] is a segmentation algorithm that only relies on point coordinates. It first generates an error signal to indicate the deviation of each point from its expected location by using some interpolation kernel. Then the trajectory is partitioned into segments by choosing a threshold value in this signal. In contrast with the unsupervised SWS, a classification-aided improvement of SWS called WS-II is proposed in [3]. As with SWS, firstly WS-II uses the same procedure to generate an error signal, but secondly, WS-II uses a window-based classifier to determine if a point indicates a partitioning position. However, the segments returned by WS-II still do not contain labels. GRASP-UTS [2] is a segmentation algorithm based on the minimum description length (MDL) principle. It splits the trajectory by minimizing a cost function which considers both homogeneity within segments and separation of landmarks of segments. In summary, the main limitation with these studies is that the returned segments do not have labels regarding fishing activities. So in this work, we are not only interested in finding the partitioning positions, but also in assigning appropriate labels for the segments.

III. Proposed Methodology

The trajectory data used in this work comes from the Automatic Identification System (AIS). One trajectory object is an ordered sequence of timestamped geospatial points (p_i, t_i). Given a trajectory T of a fishing ship, this work aims to split T into a sequence of k adjacent segments $<(S_1,l_1), \ldots, (S_k,l_k)>$, where S_i is a continuous sequence of points in T, and l_i is the label of segment $\in$ {fishing, non-fishing}.

The methodology proposed in this work comprises two stages Fig. 2. Firstly, we introduce a technique based on visualization to help design features characterising movement patterns. Secondly, an algorithm is proposed to split each trajectory into fishing and non-fishing segments by combining a pre-trained classifier and the run-length encoding technique.

A. Feature Design by $TPoSTE$

In this section, we introduce a visual exploration technique called $TPoSTE$ to help design features characterising different movement patterns. The motivation is that a ship's movement during fishing can be complex and irregular, making it non-trivial to devise effective spatiotemporal criteria for the segmentation purpose. Given a trajectory T, $TPoSTE$ is carried out in three steps:

1) choose spatiotemporal events of interest in T.
2) for each event, find all occurrences of it in T and plot those occurrences using the start time and the end time. Occurrences for different events are plotted parallel.
3) observe and gain some insights which can help design useful features. If needed, go back to the first step and choose new events of interest.

It is worth mentioning that the idea of using visual analysis to guide the feature engineering has been in use in the visualization community, e.g., [14].

Next, we use an example to illustrate how $TPoSTE$ can be used to design features characterising movement patterns.

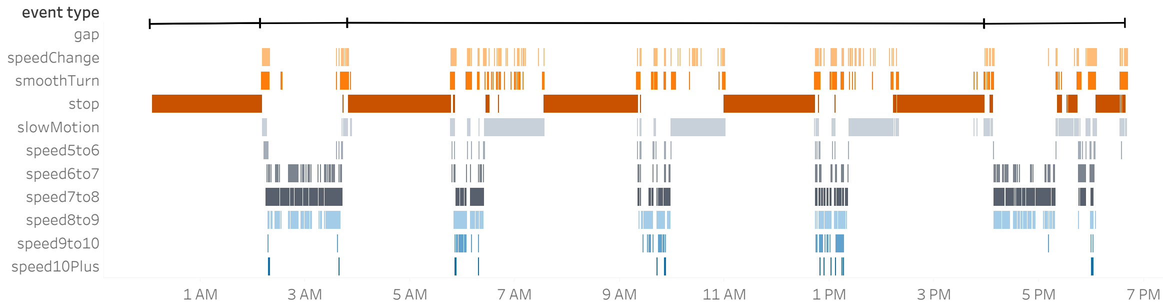

Fig. 3. Temporal profiling of events in a trajectory on Nov 16, 2021 by the ship with mmsi#220051000.

Step #1. In this step, we have chosen events related to speed, heading change, and temporal gap. Among them, speed and heading change have been used a lot in the literature [5], [4] to analyse ship behaviors. The temporal gap is considered here because AIS signals may be lost due to lack of coverage in some regions or manually switching-off of AIS.

In this work, based on multiple trials, a temporal gap event means that between two consecutive observations a gap of 30 minutes or more is observed. The $smoothTurn$ event means that the heading at a point (p_i, t_i) has deviated from the mean heading of recent movement by an angle larger than a threshold, e.g., 10°, where the mean heading is calculated based on the k preceding points (here k=7), similar to [15]. We also introduce several events using speed ranges (in knots/hour): the $stop$ event for speed range [0,1), the $slowMotion$ event for speed range [1,5), then six events $speed5to6$, $speed6to7$, $speed7to8$, $speed8to9$, $speed9to10$, $speed10Plus$ that are defined analogously. Finally, the $speedChange$ event means that the speed at a point has deviated from the mean speed in the previous m=7 points by more than 25%.

Step #2. In this step, all occurrences of the above events are identified in a trajectory, and each occurrence is associated with its start and end time, and added to the plot. The $speedChange$ events are not plot when the speed is below 1 knot/h, because in such case a ship's location has much noise, making the heading change unreliable. Fig. 3 illustrates this step when applied to the trajectory in Fig. 1. Each rectangle in Fig. 3 represents an occurrence of the corresponding event. Different events are drawn in parallel to account for the fact that multiple events can happen simultaneously.

Step #3. This is the most important step where we try to gain some insights from the result in Step #2. These insights will help us design features to distinguish movement patterns.

The trajectory in Fig. 1 shows four main parts, as captured in Fig. 3 by the four black line segments at the top. Firstly there is a two-hour anchorage at midnight, where the ship's location remained almost the same. So a long $stop$ event was shown at the left in Fig. 3. Secondly, the ship left the port and steamed to the fishing ground. It took about 1.5 hours to arrive there. The main characteristic in this part was that the ship sailed mostly at relatively high speeds (6-7 knots/h). Also, there were fewer $smoothTurn$ and $speedChange$ events in between than the two ends of this part.

The third part was a twelve-hour fishing activity in the fishnig ground until late afternoon. Fig. 3 shows that there were some recurrent patterns during this period, which were evidenced by the three petal-like rings in Fig. 1. Within each ring, the ship first stopped for about two hours (maybe for some preparatory work). Then it began to sail at speeds between 6 and 10 knots/h, and one noticeable difference was that there seemed to be more $smoothTurn$ and $speedChange$ events during this time compared to steaming in the second part. Next, the ship slowed down to between 1 and 5 knots/h and returned to the origin to finish the ring. Finally, the ship prepared and started its next ring. Overall, the ship took about 4 hours to complete each ring.

Lastly, the ship returned to the port. It first sailed at speeds between 6 and 9 knots/h, and again we notice that there were fewer $speedChange$ and $smoothTurn$ events during this period compared to that during fishing. Interestingly, then the ship turned a right angle and slowed down to below 5 knots/h for some unknown reason when it came close to the port. Finally, the ship arrived at the port and anchored again.

As can be seen, the gap event did not happen at all during the whole trajectory, and this fact suggests that the gap event is less useful for our analysis than other events. Also, we see that a ship tends to move slowly during fishing. However, an anchored ship also exhibits similar behavior. Since we care more about ship activities in the open sea, these anchorage points need to be removed to facilitate later analysis.

On the one hand, the above analysis shows that fishing activities can be complex and thus it is challenging to design effective spatiotemporal criteria to detect them. On the other hand, Fig. 3 indicates that movement patterns can only be captured by considering a context that contains sufficient information for profiling ship behaviors. Those two aspects inspire us to resort to an approach that can learn fishing patterns and this approach will be presented later in III-B.

Based on the above analysis, we have designed the following features to learn fishing and non-fishing patterns.

- *Turn Frequency* is the average number of $smoothTurn$ events that happen during a time period.
- *Speed-Change Frequency* is the average number of $speedChange$ events that happen within a time period.

For each of the remaining eight speed-related events, we design the following features.

- *Average Duration* is the average time duration of all occurrences of an event.
- *Duration Ratio* is the ratio of the total duration of all occurrences of an event to the duration of a trajectory.
- *Spanning Ratio* is the ratio of the temporal gap between the start time of the first occurrence and the end time of the last occurrence to the duration of a trajectory.
- *Frequency of Events* is the average number of occurrences of an event during a time period.

B. Window-Based Trajectory Segmentation using Run-Length Encoding: WBS-RLE

The idea of *WBS-RLE* is to learn fishing and non-fishing patterns from labelled trajectory data and use learned patterns to detect fishing activities on new trajectory data. Specifically, *WBS-RLE* comprises three steps. Firstly, it trains a classifier on labelled trajectory data. Secondly, a sequence of windows are generated from new trajectories and these windows are classified using the classifier in the first step. Finally, to detect complete fishing activities, a run-length encoding technique is used to combine close fishing windows into a fishing activity.

Window Generation. Instead of working with individual points as in [10], [9], *WBS-RLE* uses a window as its analysis unit. A window in this work is a trajectory segment that has a time duration larger than a threshold and contains at least a specified number of points. We also require that two adjacent windows have an overlap of a given percentage of the points in the window. To learn fishing and non-fishing patterns, windows are generated from some labelled trajectory data, then features are extracted from those windows and fed into some classification model to learn a classifier, which then can be used on windows from new data.

Run-Length Encoding for Detection of Complete Fishing Activity. Each window from new trajectory data can be labelled as fishing or non-fishing by applying the pre-trained classifier, so a sequence of labelled windows can be obtained. However, fishing activities can be up to hours and even days, and it is likely that multiple fishing windows actually belong to one complete fishing activity. So the last step is to combine those close fishing windows into complete fishing efforts. To this end, we adopt the run-length encoding technique to count the number of consecutive occurrences of fishing and non-fishing windows. Thus we can get a sequence of alternating counts like $\ldots, a_{fishing}, b_{non-fishing}, c_{fishing}, \ldots$, where a, b, c are the counts of windows. From the perspective of run-length encoding, a complete fishing activity A in this work is defined as a maximal subsequence of counts satisfying:

- A starts and ends with fishing counts.
- each triplet $<a_{fishing}$, $b_{sailing}$, $c_{fishing}>$ in A fulfills $a \geq b$ and $b \leq c$ to correct occasional classification errors.

So the final complete fishing activities are detected using the above conditions and a trajectory can be split at boundary points of fishing activities to generate non-fishing activities.

IV. Experiments

A. Dataset

In the literature, many algorithms [2], [7], [3] have been evaluated on a fishing dataset introduced in [2]. The main problem with this dataset is that the average sampling rate between two points is 105 minutes [16], so it is very sparse and may lose important information of movement patterns. Moreover, it is a small dataset, limited to 5,190 points, so it can not support fine-grained analysis of fishing trajectories.

Therefore, we chose to use another dataset that has a higher sampling frequency, and it is publicly available from the Danish Maritime Authority [2]. For our study, we used one-week AIS data between Nov 14, 2021 and Nov 20, 2021 around Danish waters. Because this work concerns fishing activities, we only retain AIS data generated by fishing ships.

Pre-processing. Since AIS data comes with several quality issues [17], for each ship we removed the AIS records with duplicate locations, and then averaged the coordinates if there were multiple AIS records for a single timestamp. Then we manually labelled 128 trajectories, which contain 1,080,220 points and have an average sampling interval of 10.63 seconds.

B. Classifier Training

To train the classifier for fishing and non-fishing patterns, we prepared a training dataset including 2,406 fishing windows and 1,504 non-fishing windows from 31 of the 128 trajectories. These windows were created with the following parameters: $size_w$ = 300, t_w = 1 hour, and $ratio$ = 5/6. The value of t_w is determined as follows. Firstly, it can not be too large, otherwise, fishing and non-fishing activities are likely to be mixed. Secondly, it can not be too small, otherwise a window will not contain enough information to capture movement patterns. The value of $ratio$ determines the expected number of windows a point belongs to, e.g., a ratio of 5/6 means that each point is expected to be included in 6 windows. A larger ratio can increase the robustness of our algorithm but also implies a higher computational cost. So in some sense, $ratio$ and t_w determine the granularity of our analysis.

Using this labelled dataset, we trained a random forest classifier with 50 trees having a depth of 10, and Table I shows the top-5 important features. Surprisingly, all of them are speed-related features. Also it suggests that it was a good decision to have finer speed ranges. Although some features (e.g., turn frequency) chosen by *TPoSTE* were not assigned high importance, $TPoSTE$ is still a helpful tool, because it encourages users to devise and try various features of interest.

C. Results

In this section, we report the results by applying WBS-RLE to the remaining 97 trajectories. Each of them has three ground truth segments: sailing to fishing ground, fishing, returning to harbour. For comparison, we have chosen three state-of-the-art methods: CB-SMoT [4], Warped K-Means [6], SWS [7]. Their implementations are publicly available from Github[3].

[2]https://web.ais.dk/aisdata/
[3]https://github.com/metemaad/TrajSeg

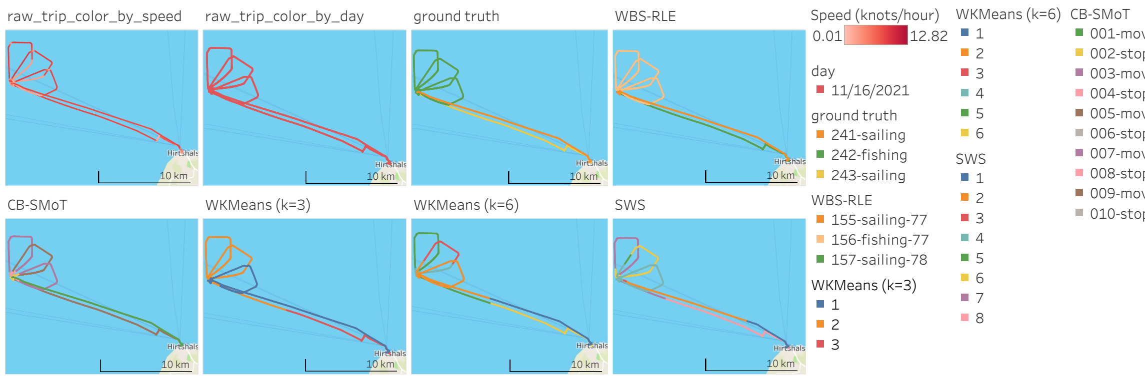

Fig. 4. Segment results of four algorithms on the trajectory #220051000-2

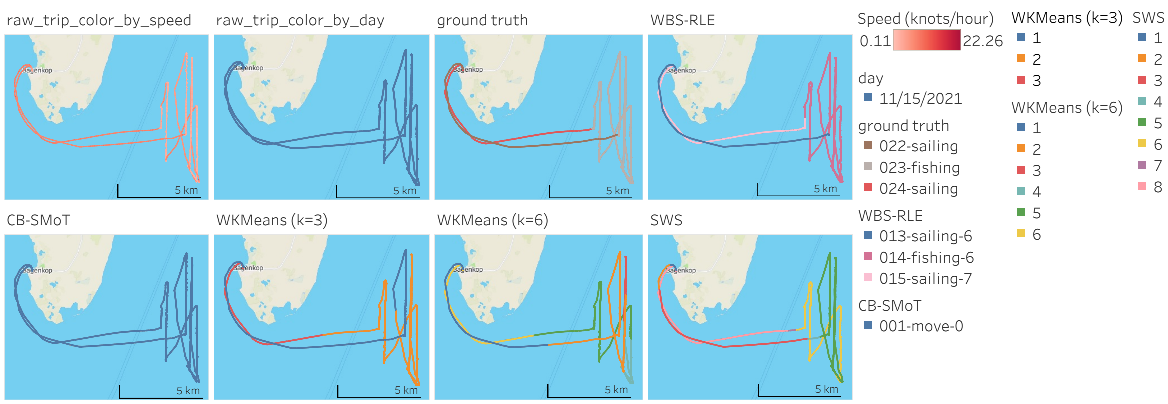

Fig. 5. Segment results of four algorithms on the trajectory #219002136-3

TABLE I
FEATURE IMPORTANCE

feature	importance	feature	importance
slowMotionTimeTotalAvg	0.382	slowMotionTimeAvg	0.201
speed7to8NumberAvg	0.101	speed7to8TimeSpanAvg	0.056
stopTimeSpanAvg	0.039		

TABLE II
PERFORMANCE OF THE FOUR ALGORITHMS ON TWO TRAJECTORIES

trajectory ID	method	purity	coverage	harmonic mean	# of segments
220051000-2	WBS-RLE	0.832	0.965	0.894	3
	CB-SMoT	0.998	0.634	0.776	10
	WKMeans (k=3)	0.888	0.826	0.856	3
	WKMeans (k=6)	0.962	0.638	0.767	6
	SWS	0.985	0.650	0.783	8
219002136-3	WBS-RLE	0.908	0.991	0.948	3
	CB-SMoT	0.861	1	0.925	1
	WKMeans (k=3)	0.903	0.861	0.882	3
	WKMeans (k=6)	0.984	0.644	0.778	6
	SWS	0.998	0.732	0.844	8

TABLE III
AVERAGE PERFORMANCE ON 97 TRAJECTORIES

method	purity	coverage	harmonic mean	# of segments
WBS-RLE	0.890	0.974	0.927	2.670
CB-SMoT	0.859	0.885	0.859	5
WKMeans (k=3)	0.878	0.840	0.855	3
WKMeans (k=6)	0.932	0.619	0.741	6
SWS	0.954	0.759	0.837	9.855

Parameter Setting. The parameters for the algorithms were determined as follows. Since many existing work (e.g., [17]) consider ships to be still when the speed is below 1 knot/h, the speed threshold in CB-SMoT was set as 1 knot/h and the neighborhood size was set as 500 meters. Because there are 3 segments in each of the 97 trajectories, we tried two k values (3 and 6) for Warped K-Means. As recommended in [16], we adopted the linear regression kernel and a window size of 7 for SWS, and the percentile in SWS was set as 99.9 because our dataset had a much higher sampling frequency. For WBS-RLE, we additionally required that each returned fishing activity must have a duration of at least 2 hours, given the fact that fishing activities usually last several hours.

Evaluation Metric. For performance evaluation, we adopted two widely-used indices proposed in [2]: purity and coverage. Simply speaking, for each returned segment r, purity measures the largest proportion of points in r that indeed belong to a ground truth segment, whereas for each ground truth segment g, coverage measures the largest proportion of points in g that are included in a returned segment. As in [16], the harmonic mean of purity and coverage was also used because we want to achieve both high purity and high coverage. Finally, since one important aspect of our work is to return complete fishing activities, we also report the number of returned segments.

To illustrate the results, Fig. 4 and Fig. 5 show the returned segments by different algorithms on two trajectories #220051000-2 and #219002136-3, and the evaluation metrics for these two trajectories are shown in Table II.

- For the trajectory #220051000-2, WBS-RLE outperformed all competitors and had the highest harmonic mean. In terms of purity, all other algorithms were higher than WBS-RLE because purity tends to be high when there is a large number of returned segments. However, it can be seen that CB-SMoT, WKMeans (k=6) and SWS had a much smaller coverage than WBS-RLE and WKMeans (k=3), because they returned too fragmented segments as shown in Fig. 4. In addition, Fig. 4 shows that WBS-RLE returned more meaningful segments which are close to the ground truth.
- For the trajectory #219002163-3, WBS-RLE again achieved the highest harmonic mean. Interestingly, CB-SMoT returned only one segment as its result because there were not many consecutive points in this trajectory that had a speed smaller than 1 knot/h. Also, Fig. 5 shows that the segments returned by WBS-RLE were more visually appealing and closer to the ground truth segments.

Table III shows the average of metrics on all the 97 trajectories. As before, WBS-RLE had the highest harmonic mean, and its coverage was also the highest. However, one weakness of WBS-RLE was that it may return the entire trajectory as one fishing activity when the sampling frequency on some local parts was low or a ship returned to harbor at low speeds. This weakness was indicated by an average of 2.67 segments by WBS-RLE. In fact, WBS-RLE returned 1 or 2 segments for 23 of the 97 trajectories. Nevertheless, the average number of segments by WBS-RLE was still the closest to that in the ground truth (i.e., 3) among all algorithms.

V. Conclusion

The sustainable use of marine resources and combat against IUU fishing is becoming an increasing concern in recent years. In this work, we studied the problem of semantic segmentation of AIS trajectories, and it aims to detect fishing activities as completely as possible. Our methodology includes two modules. Firstly, we proposed a technique based on visualization to help design features that can capture different movement patterns. Secondly, we proposed a window-based segmentation algorithm that can utilize both machine learning and run-length encoding to return semantically meaningful activities. The performance of our methodology was evaluated on a real dataset both quantitatively and visually, and results showed that our method outperformed state-of-the-art methods and returned more meaningful segments. Our methodology can be used to discover potential IUU fishing activities, also it can be used to aid the annotation of large AIS trajectory datasets. For future research, we plan to investigate the segmentation of trajectories of other ship types and activities, e.g., ferries, transit through a canal, etc.

Acknowledgment

This work is funded by the Horizon EU DEDS project, under the Marie-Slodowska Curie grant agreement No 955895.

References

[1] B. Chuaysi and S. Kiattisin, “Fishing vessels behavior identification for combating iuu fishing: Enable traceability at sea,” *Wireless Personal Communications*, pp. 1–23, 2020.

[2] A. Soares Júnior, B. N. Moreno, V. C. Times, S. Matwin, and L. d. A. F. Cabral, “Grasp-uts: an algorithm for unsupervised trajectory segmentation,” *International Journal of Geographical Information Science*, vol. 29, no. 1, pp. 46–68, 2015.

[3] M. Etemad, Z. Etemad, A. Soares, V. Bogorny, S. Matwin, and L. Torgo, “Wise sliding window segmentation: A classification-aided approach for trajectory segmentation,” in *Canadian Conference on Artificial Intelligence*. Springer, 2020, pp. 208–219.

[4] A. T. Palma, V. Bogorny, B. Kuijpers, and L. O. Alvares, “A clustering-based approach for discovering interesting places in trajectories,” in *Proceedings of the 2008 ACM SAC*. ACM, 2008, p. 863–868.

[5] J. A. M. R. Rocha, V. C. Times, G. Oliveira, L. O. Alvares, and V. Bogorny, “DB-SMoT: A direction-based spatio-temporal clustering method,” in *2010 5th IEEE International Conference Intelligent Systems*, 2010, pp. 114–119.

[6] L. A. Leiva and E. Vidal, “Warped k-means: An algorithm to cluster sequentially-distributed data,” *Information Sciences*, vol. 237, pp. 196–210, 2013.

[7] M. Etemad, A. Soares, E. Etemad, J. Rose, L. Torgo, and S. Matwin, “Sws: an unsupervised trajectory segmentation algorithm based on change detection with interpolation kernels,” *GeoInformatica*, vol. 25, no. 2, pp. 269–289, 2021.

[8] M. Buchin, A. Driemel, M. Van Kreveld, and V. Sacristán, “Segmenting trajectories: A framework and algorithms using spatiotemporal criteria,” *Journal of Spatial Information Science*, no. 3, pp. 33–63, 2011.

[9] F. Natale, M. Gibin, A. Alessandrini, M. Vespe, and A. Paulrud, “Mapping fishing effort through ais data,” *PloS*, vol. 10, no. 6, 2015.

[10] G. Rovinelli, S. Matwin, F. Pranovi, E. Russo, C. Silvestri, M. Simeoni, and A. Raffaetà, “Multiple aspect trajectories: a case study on fishing vessels in the northern adriatic sea.” in *EDBT/ICDT Workshops*, 2021.

[11] F. Mazzarella, M. Vespe, D. Damalas, and G. Osio, “Discovering vessel activities at sea using ais data: Mapping of fishing footprints,” in *17th International conference on information fusion (FUSION)*. IEEE, 2014, pp. 1–7.

[12] S. Spaccapietra, C. Parent, M. L. Damiani, J. A. de Macedo, F. Porto, and C. Vangenot, “A conceptual view on trajectories,” *Data & knowledge engineering*, vol. 65, no. 1, pp. 126–146, 2008.

[13] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, “A density-based algorithm for discovering clusters in large spatial databases with noise.” in *kdd*, vol. 96, no. 34, 1996, pp. 226–231.

[14] G. Andrienko, N. Andrienko, G. Anzer, P. Bauer, G. Budziak, G. Fuchs, D. Hecker, H. Weber, and S. Wrobel, “Constructing spaces and times for tactical analysis in football,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 4, pp. 2280–2297, 2021.

[15] K. Patroumpas, E. Chondrodima, N. Pelekis, and Y. Theodoridis, “Trajectory detection and summarization over surveillance data streams,” in *Big Data Analytics for Time-Critical Mobility Forecasting*. Springer, 2020, pp. 85–120.

[16] M. Etemad, “Novel algorithms for trajectory segmentation based on interpolation-based change detection strategies,” Ph.D. dissertation, Dalhousie University, 2020.

[17] K. Patroumpas, E. Alevizos, A. Artikis, M. Vodas, N. Pelekis, and Y. Theodoridis, “Online event recognition from moving vessel trajectories,” *GeoInformatica*, vol. 21, no. 2, pp. 389–427, 2017.