Segmentation Optimization in Trajectory-Based Ship Classification

Daniel Amigo(✉), David Sánchez(✉), Jesús García, and José Manuel Molina

Group GIAA, University Carlos III of Madrid, Madrid, Spain
{damigo,davsanch,jgherrer,molina}@inf.uc3m.es

Abstract. An analysis over trajectory segmentation techniques is carried out by the study of the different algorithms and the experimentation over a ship classification problem, which use a data preparation and classification system used in previous works. With the data preparation, the system handles real-world Automatic Identification System (AIS) data, cleaning wrong measurements and smoothening the trajectories by the application of an Interacting Multiple Model (IMM) filter. Also applies some balancing algorithms to address the lack of an equal distribution among classes. To correctly evaluate the classification with the imbalanced data a multiple objective analysis is proposed to consider the minority class and the global accuracy. Over that multi-objective analysis, different segmentation algorithms and its variations are tested to analyze the influence of them into the classification problem. The results show a Pareto front with different viable solutions for the proposed multi-objective problem, without a dominant algorithm over rest of the tested segmentation algorithms.

Keywords: AIS data · Class imbalance · Kinematic behavior · Ship classification · Track segmentation

1 Introduction

The maritime surveillance systems are an essential element for the protection of the seas, ensuring the safety of maritime transport and security of citizens. Detecting and locating vehicles is a solved problem using multiple technologies, but classifying the type of vessel is more challenging, which is an essential element for decision-making in maritime surveillance systems. Technologies such as AIS [1] provide information that allows the target identification, however as they work collaboratively the information is not always reliable, as it is susceptible to manipulation.

The problem of this study is the classification of trajectories to obtain the type of ship based on kinematics data that model its behavior. This is an extension of a previous study [2, 3], where the problem was defined and main subprocesses identified. These first approaches concluded that it was necessary to specifically analyze the impact of each subprocess on the classification. Thus, the objective of this paper is to study the impact of segmentation on the final performance, observing the variation compared the fixed-size

Á. Herrero et al. (Eds.): SOCO 2020, AISC 1268, pp. 540–549, 2021.
https://doi.org/10.1007/978-3-030-57802-2_52

segmentation initially proposed. To achieve it, more complex segmentation techniques are studied and analyzed, allowing variable size segments that can be better adjusted to the ships' motion. To proceed from the sensor detections to the ship classification, it is required a system that performs different processes on the data. This system has been developed in previous works [2, 3], being necessary within this study the analysis of the segmentation process.

The system used has several processes, starting from the data preparation to clean real-world data problems, an IMM filter is used to reduce the noise by smoothing the target trajectory. The proposed step is the segmentation of trajectories, splitting the original track by applying different criteria (uniform length, shape or direction preserving…) and then a process handles of the data imbalance, since the ship types are not distributed in a homogeneous manner (neither in trajectories or segments). Finally, the last process is classification, by using different algorithms applied to track segments to predict the ship type. Specifically, the objective is to determinate the membership in the fishing class, which is the minority in the used dataset. This classification process requires a prior sub-process that computes representative features from each trajectory segment, these will be variables used to model the behavior of the ship. Although other variables related to the trajectories context could provide useful information to classify them, the proposed system seeks to avoid this type of information, because it aims to find a system based on as little information as possible, focusing only on the track kinematics, which could be improved later by including the context information.

The experiments compare various segmentation techniques with respect to the original segmentation (fixed length). The results show the trade-off between accuracy and imbalance of classification so there is not an absolute optimal solution, but makes it clear the multi-objective nature of the problem, and solutions show a Pareto front.

This paper is organized as follows: In Sect. 2 the state-of-art methods in segmentation of maritime vehicles tracks are analyzed. In Sect. 3 is explained the process necessary to the trajectory-based classification in Sect. 4 there is the explanation of all the texted segmentations while in Sect. 5 results of the work are shown. Finally, the conclusions and perspectives for future works are presented in Sect. 6.

2 State of the Art

The state of the art covers previous works on two main problems: trajectory classification and trajectory segmentation.

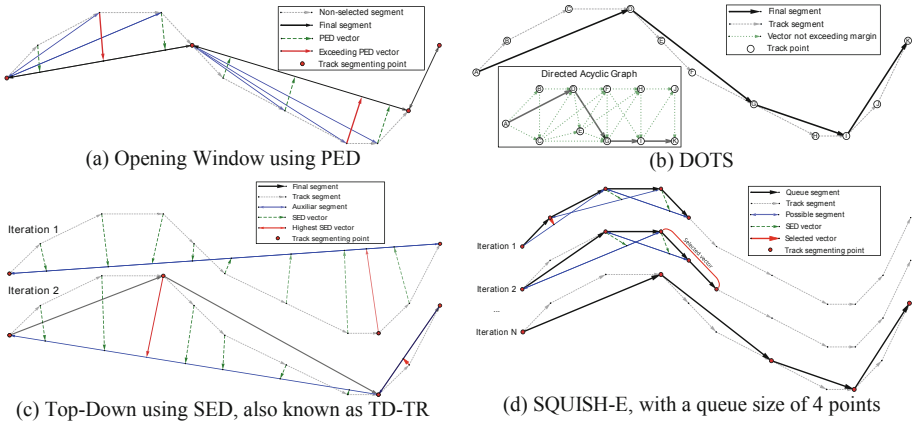
A basic problem for trajectory classification is the feature extraction to infer intelligence from the available information. For example, these recent studies [4–6] perform a feature extraction on the trajectory of the ship to determine its behavior. This feature extraction is not adequate for a problem where long-duration trajectories or very heterogeneous mixture of trajectories appear.

As an alternative, feature extraction can be applied on each segment instead of the whole track in order to extract more precise information for the classifier. There are researchers [7] who perform a segmentation before classification, but they use their own segmentation technique very specific to their problem. Alternatively, this paper experiments with both classical and recent segmentation techniques to analyze how they

influence the problem of classification trajectories. Note also that all these papers use context information, making them incomparable with the present proposal.

The field of trajectory segmentation has several approaches [8], one of them is the compression algorithms, which identify the key-points of the trajectory and use them to generate the segments. Segments are generated according to different conditions, e.g. time gaps, trajectory shape or its context. Also, they can be categorized according to whether they need the entire track (offline) or they can run in real time (online).

The simplest approach to segmentation is uniform sampling, which cut the track into segments of uniform size [9] (the approach used by default in the previous works). This paper explores segmentation algorithms according to the trajectory shape, generating segments that minimize error with respect to the trajectory. In Fig. 1 illustrates several segmentation algorithms achieving different outputs on the same track.



(a) Opening Window using PED

(b) DOTS

(c) Top-Down using SED, also known as TD-TR

(d) SQUISH-E, with a queue size of 4 points

Fig. 1. Example of how several track segmentation algorithms work

The classic algorithms for segmentation are:

- Opening Window (OPW) [10]: It generates variable size segments by setting the start of the track and including points in the window until an error threshold is exceeded. When this threshold is exceeded, as is shown in Fig. 1(a), the current segment is closed, and the window is restarted, following this process until the end.
- Top-down [11]: It starts with a segment that covers the entire trajectory and divides it recursively at the point where the error is highest, as shown happening twice in Fig. 1(c). This process continues until the selected error measurement is below the threshold for all points.
- Bottom-up: The inverse process to Top-Down. It starts with small segments unifying them when the error is the smallest, until cannot be unified anymore.

These algorithms calculate the segment error in relation to the trajectory by using the Perpendicular Euclidean Distance (PED) of each point. A big improvement is to

use instead of PED, the Synchronized Euclidean Distance (SED) [10], which take into consideration track point timestamp with regard to the segment total time.

Based on the previous classic approaches there are many other algorithms that seek a better performance when performing the segmentation, like:

- SQUISH-E [12]: It works by using a queue of fixed size, adding points to it and in each iteration eliminating the one with the smallest SED error. Figure 1(d) shows this procedure, checking in trios the less relevant point and removing it from the queue. This algorithm uses two parameters for shaping the resulting segment: λ guarantees a compression ratio of the track, while μ indicates the maximum SED error.
- MRPA [13]: It works by approaching the track based on a bottom-up multiresolution approach, using an accumulated variation of the SED criterion (ISSED).
- DOTS [14]: This algorithm performs a variation to allow online running of the MRPA. It uses a DAG (Directed Acyclic Graph) to describe all potential segments of the trajectory, as can be shown in Fig. 1(b).

3 Ship-Type Determination Using Binary Classification

This section provides a brief explanation of the original system used on the general problem, summarizing its main subprocesses, starting from the input data up to the classification algorithms. The system was detailed in [2, 3].

The first step is cleaning the raw data from sensors. In this case, the available data is from AIS sensor. It provides kinematic data of ships integrated with additional information such as the ship type, which is used here to train the classifier. Specifically, the chosen repository is the one provided by the Danish Maritime Authority [15], in which there is a recompilation of daily AIS contacts since 2006. Dealing with real-world raw data requires a strong pre-processing which is critical for final performance, removing inconsistencies, null, wrong, and noisy values. These problems are generated by malfunction of AIS transmitters and human errors. The measurement noise taken by the sensor can either be outliers, directly detectable evaluating the offset in GPS coordinates, or small noises that can be smoothed by a filtering algorithm. In the proposed system an IMM filter has been implemented to smooth the noise, configured with two Extended Kalman Filters as modes of prediction for ship trajectories: the first one for linear movements and low prediction noise and the second one to model the movements that would be considered noisy (speed variations, turns, …).

Prior to classification, its necessary a process to address the unbalance problem present in this domain due to the lack of an equal distribution among classes. For instance, long and frequent trajectories of cargo and passenger vessels populate the training data sets and bias the classification models towards these categories reducing the representation of other ones, like the fishing vessel category. To solve the problem, the system implements oversampling and undersampling techniques, which adjust the amount of data of each class by adding or removing instances [16]. The experimentation uses the original imbalanced dataset, and two balanced datasets: one using random undersampling, randomly removing instances of the majority classes, and another using the SMOTE algorithm [17], already used for track classification [5], oversampling the

minority class by creating new artificial samples. The classification is based on the following features generated from the track points contained in the segments:

- Course variation: describing turnarounds between track points.
- Distance: characterizing movement range and complexity between track points.
- Speed: characterizing the movement velocity between track points.
- Time between measures: considering the time gaps between track points.
- Speed variation: describing acceleration and deceleration between track points.

Because the possible difference in the number of measures between segments, is necessary to make those kinematic variables suitable as a classification input. The following statistical measures are applied to aggregate all the segment track points: the mean, maximum, minimum, mode, standard deviation and three quartiles. Also, the total time of the segment is included to support the time gaps variables.

The classification problem considered in this work is predicting when a vessel is of fishing type and when it is not, i.e. a binary classification problem. Common classification algorithms in binary problems as the Support Vector Machine (SVM) and the decision tree algorithm are chosen, looking to keep the importance on the segmentation problem by using simple and well-known techniques but able to perform it.

To evaluate the results obtained by the classification we must consider two main factors, the accuracy of the general classification and the specific accuracy on the minority class (fishing), which is affected by the imbalance in the training process. Therefore, along with the classification accuracy, the F-measure metric [18] is considered to assess both effects. The simultaneous evaluation of both metrics prevents the domination of the classification accuracy by the effect of majority class. Besides, the presence of these two metrics makes the problem multi-objective, allowing to observe the Pareto's front when displaying the results from different algorithms and their parameters.

4 Trajectories Segmentation

This section presents the different experiments to be carried out using the track segmentation algorithms. Each algorithm has different parameters to set its functionality depending on the problem. In this case, as the configuration of each algorithm is not trivial with respect to its impact on the classification, different experiments are performed, varying from each of the parameters, allowing an analysis of the impact of each of them. A summary of the variations of each algorithm is shown in Table 1 and a detailed explanation of the 196 experiments tested in this paper is given below.

The base case used in previous works uses a uniform segmentation of 50 measures (around 9 min). For comparison, tests of 10 and 20 values are performed as well.

Opening window (OPW) has the following variants from its base implementation:

- The cut-off criterion: whether it occurs at the point where the window has exceeded the error (NOPW), and whether it is done at the previous point (BOPW) [10].
- Different error evaluation functions: PED or SED ("_TR", meaning Time-Ratio [10]). Three error values are tested to each function: 20, 30 and 50 m.

Table 1. Segmentation algorithms variations

Base algorithm	Variation (if any)	Error function	Error value (meters)	Minimum size	Compression rate
Uniform Segment		PED	-	0 10 20 50	-
OPW	BOPW NOPW BOPW_TR NOPW_TR	PED SED	20 30 50		
TopDown	DP TD_TR				
BottomUp		PED		-	1, 5, 10
SQUISH-E		SED			
DOTS		ISSED	100 500		-
MRPA					

- To ensure that the segments are generated with a minimum length, favoring the classification. A minimum segment size its tested with 0, 10, 20 and 50 points.

The Top Down algorithm has variations for the error evaluation function, marked as "DP" (Douglas Peucker algorithm [11]) when it uses PED and as "TD_TR" when it uses SED [10]. These variations use the same error and minimum segment size as OPW.

Bottom Up has no relevant variations according to the error function, as only the PED error function has been used in the literature.

SQUISH-E only uses the SED, with the same three error values already listed as μ value. In addition, it has the compression parameter λ, testing 1, 5 and 10 values.

Finally, both DOTS and MRPA only vary on the error values, using 100 and 500 as values for its accumulative SED variation.

5 Results Analysis

The performed experimentation is applied over three days in July 2017 from AIS contacts off the coast of Denmark. In total, more than 30 million contacts are available as system inputs. After the cleaning process, there are 7 million contacts, divided into 39077 different tracks. These trajectories are the inputs of the segmentation stage, which results in the number of segments shown in Fig. 2.

The figure also shows a demonstration of the imbalance problem, being possible to see the difference between the fishing class and the remaining instances (non-fishing).

As mentioned, to analyze the results of the different experiments carried out, the accuracy and F-measure are displayed together as a multi-objective problem, considering the total accuracy and the problem imbalance problem at the same time. In the Fig. 3, it can be seen the distribution of values of the accuracy and F-measure corresponding to different variations of the classification and balancing algorithms. The Pareto front is formed for those non-dominated solutions, i.e., those with no other solutions with higher values in the two metrics simultaneously. In the figure, this front is formed by the solutions appearing in the upper-right corner.

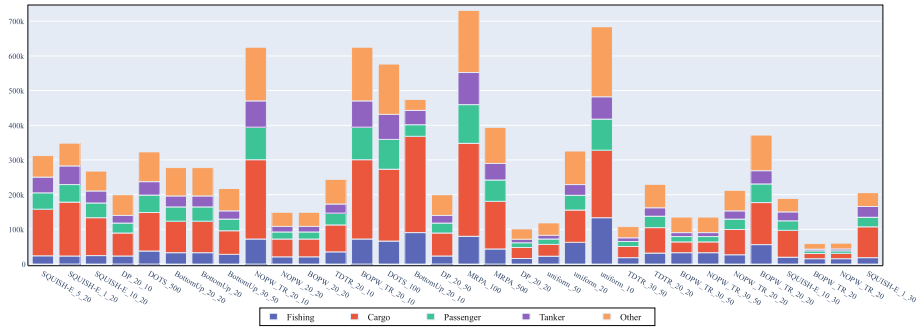


Fig. 2. Number of segments of the main ship types

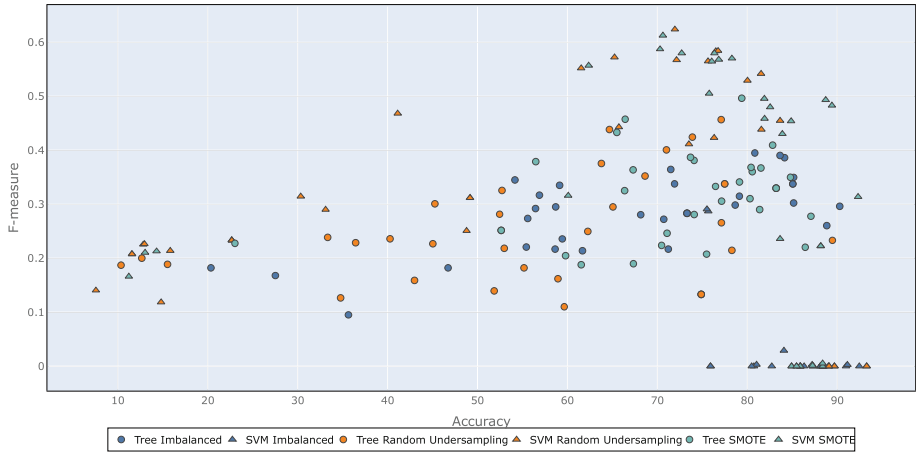


Fig. 3. Classification results for the different proposed variations

It can be appreciated how the SVM has results that are usually better with respect to accuracy, but in return it may have a worse performance when considering the class imbalance. That effect is produced because it is a boundary-based algorithm and has a trend to misclassify the minority class if it has a low impact in the total accuracy. This is especially noticeable in the imbalanced classification, which shows in many cases a zero value for F-measure (i.e., all samples of the minority class misclassified).

The decision trees have more moderate results, which do not stand out so much in the accuracy but in return they get better results in the F-measure. However, the front is clearly dominated by the SVM with balanced data sets, these although still have executions that demonstrate little success in the problem of the imbalance but also have the executions located in the front.

The most notable of these are the SVMs that operate on a balanced data set using SMOTE, although the random undersampling also have pareto front executions. To put the results in perspective, Fig. 4 shows all the segmentation algorithms executed by SVM applied on the SMOTE balanced data set.

Fig. 4. SVM classification result for the segment variation in SMOTE balanced dataset

The figure not only shows the results of the accuracy but also the results for the F-measure which is not so positive since the most complex segmentations usually have slightly lower results in that metric.

There is no case that stands out especially from the rest, since when talking about a multi-objective problem between unbalance metrics and classification accuracy there is no algorithm that is especially good in both.

Being a point to emphasize that the best algorithms in one of the objectives clearly obtain their improvement when getting worse in the other one, an example would be the SQUISH-E with 20 error value and 5 compression parameter that obtains the best accuracy although its metrics are far below other algorithms. There is also the opposite case with the opening window algorithm, in which the best F-measure show an accuracy 20 points below that obtained by the specified SQUISH-E.

Regarding the higher complexity of the segmentation algorithms we can see how generally the segmentation algorithms that give better results when performing the compression of trajectories (SQUISH-E, MRPA, DOTS) do not ensure a better result within the proposed classification problem. Most of their executions seem to have good accuracy but not all of them good results in the F-measure used for the imbalance problem. In fact, one of the results belonging to the front and that therefore could be considered as one of the best, is obtained by the most basic segmentation algorithm, the uniform sampling with a size of 50.

Another aspect to consider is that the parameters introduced in the different segmentation algorithms influence the results variation, since the different executions of the same algorithm show very different results. For example, with the SQUISH-E algorithm, is possible to observe different results: one with the best accuracies, other with very poor results and another clearly within the Pareto front, achieving one of the best values within the two objectives with an accuracy close to 90% and balancing metrics only about 10 points below the best. Even if there is no absolute solution that meets the

two proposed objectives, there is a set of solutions located on the Pareto front that are valid solutions, being better in one or the other objective.

6 Conclusions and Perspectives

In the study, the impact of segmentation on the classification results have been analyzed, being possible to appreciate as the most advanced algorithms usually provide better results in accuracy objective. However, the segments provided by these algorithms do not ensure good results in the second objective proposed, which is related to the performance with the minority class, due to the high imbalance in the data set. That said, the results show a Pareto front with different solutions that work for the two objectives imposed within the multi-objective problem

As a conclusion, it is very important the quality of the segments within the proposed process since there are trajectories with more measurements than others which create more segments with certain segmentation algorithms, affecting the classification. Also, by classifying segments it is possible to introduce noise with non-representative segments to its class (e.g. a ship departing from a port).

The SVM algorithm has demonstrated that it has the capacity to obtain good results for the classification, however it has a clear tendency towards the trivial solution, harming the minority class to obtain good results when maximizing the majority class.

Both classification algorithms are representative and responsive to the analyzed balancing algorithms. The main point of improvement is the testing of new segmentation or classification algorithms that achieve a better separation of instances, particularly those that can benefit most from the segments. Also, the application of the proposed method can approach other similar problems where classification is performed based on kinematic information of trajectories. For example, a classification oriented on pedestrian traffic could ensure safety (pickpocket identification), or the application in air traffic can allow flying mode identification thanks to the track segments adaptability.

Acknowledgement. This work was funded by public research projects of Spanish Ministry of Economy and Competitivity (MINECO), reference TEC2017-88048-C2-2-R.

References

1. Tu, E., Zhang, G., Rachmawati, L., Rajabally, E., Huang, G.B.: Exploiting AIS data for intelligent maritime navigation: a comprehensive survey from data to methodology. IEEE Trans. Intell. Transp. Syst. **19**, 1559–1582 (2018). https://doi.org/10.1109/TITS.2017.2724551
2. Amigo, D., Sánchez Pedroche, D., García, J., Molina, J.M.: AIS trajectory classification based on IMM data. In: 2019 22th International Conference on Information Fusion (FUSION), Ottawa, ON, Canada, pp. 1–8. IEEE (2019)
3. Sánchez Pedroche, D., Amigo, D., García, J., Molina, J.M.: Context information analysis from IMM filtered data classification. In: 1st Maritime Situational Awareness Workshop MSAW 2019, Lerici, Italy, p. 8 (2019)

4. Kraus, P., Mohrdieck, C., Schwenker, F.: Ship classification based on trajectory data with machine-learning methods. In: 2018 19th International Radar Symposium (IRS), Bonn, pp. 1–10. IEEE (2018)
5. Zhang, T., Zhao, S., Chen, J.: Research on ship classification based on trajectory association. In: Douligeris, C., Karagiannis, D., Apostolou, D. (eds.) Knowledge Science, Engineering and Management, pp. 327–340. Springer, Cham (2019)
6. Ichimura, S., Zhao, Q.: Route-based ship classification. In: 2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST), Morioka, Japan, pp. 1–6. IEEE (2019)
7. Sheng, K., Liu, Z., Zhou, D., He, A., Feng, C.: Research on ship classification based on trajectory features. J. Navig. **71**, 100–116 (2018). https://doi.org/10.1017/S0373463317000546
8. Zheng, Y.: Trajectory data mining: an overview. ACM Trans. Intell. Syst. Technol. **6**, 1–41 (2015). https://doi.org/10.1145/2743025
9. Tobler, W.R.: Numerical map generalization. Michigan Inter-University Community of Mathematical Geographers (1966)
10. Meratnia, N., Rolf, A.: Spatiotemporal compression techniques for moving point objects. In: Lecture Notes in Computer Science (2004). https://doi.org/10.1007/978-3-540-24741-8
11. Douglas, D.H., Peucker, T.K.: Algorithms for the reduction of the number of points required to represent a line or its caricature. Can. Cartogr. **10**, 112–122 (1973). https://doi.org/10.3138/FM57-6770-U75U-7727
12. Muckell, J., Olsen, P.W., Hwang, J.-H., Lawson, C.T., Ravi, S.S.: Compression of trajectory data: a comprehensive evaluation and new approach. Geoinformatica **18**, 435–460 (2013). https://doi.org/10.1007/s10707-013-0184-0
13. Chen, M., Xu, M., Franti, P.: A fast O(N) multiresolution polygonal approximation algorithm for GPS trajectory simplification. IEEE Trans. Image Process. **21**, 2770–2785 (2012). https://doi.org/10.1109/TIP.2012.2186146
14. Cao, W., Li, Y.: DOTS: An online and near-optimal trajectory simplification algorithm. J. Syst. Softw. **126**, 34–44 (2017). https://doi.org/10.1016/j.jss.2017.01.003
15. Danish Maritime Authority: AIS Data. dma.dk/SikkerhedTilSoes/Sejladsinformation/AIS/Sider/default.aspx
16. Gosain, A., Sardana, S.: Handling class imbalance problem using oversampling techniques: a review. In: 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, pp. 79–85. IEEE (2017)
17. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. jair **16**, 321–357 (2002)
18. Fernández, A., García, S., Galar, M., Prati, R.C., Krawczyk, B., Herrera, F.: Learning from Imbalanced Data Sets. Springer International Publishing, Cham (2018)