Clustering of maritime trajectories with AIS features for context learning

David Sánchez Pedroche, Daniel Amigo Herrero, Jesús García Herrero, José Manuel Molina López

Group GIAA, University Carlos III of Madrid, Spain

Email: { davsanch, damigo, jgherrer } @inf.uc3m.es, molina@ia.uc3m.es

***Abstract*— This paper presents an analysis on Automatic Identification System (AIS) real world ship data to build a system with the capability to extract useful information for an anomaly detection problem. The study focuses on the adjustment of a clustering technique to trajectory data, specifically using a DBSCAN algorithm that is adapted by means of two approaches. On the one hand, the DTW trajectory similarity metric is used to obtain a distance between two trajectories. On the other hand, an extraction of features of interest from each trajectory allowing a summary of the trajectory in a single multidimensional instance. The results show that both approaches are feasible, although not very scalable to larger problems due to the computational complexity of the used algorithms. In addition, the study analyses possible uses of these approaches to existing data mining problems.**

***Index Terms*— AIS data, Context learning, Data mining, Trajectory clustering**

I. Introduction

The increased use of location technologies, such the global positioning satellite systems, has led to an increase in the amount of moving objects positioning data and therefore multiple options arise when it comes to extracting useful information from their movement patterns. In the maritime environment, these analyses are applied to the trajectories of ships and other maritime vehicles to extract useful information that ensure the safety and security at seas.

An example of great interest with these maritime trajectories is the detection of anomalous behaviours in such a way that it is possible to quickly react in order to mitigate the danger. Data mining techniques applied on trajectories allow finding patterns of the trajectories that represent common behaviours. In this way, anomalies can be identified by observing elements that deviate from these found patterns.

In previous studies [1], [2] the anomaly detection problem was approached from the perspective of a ship type detection problem which is achieved through a ship classification based on the kinematic information of the ship trajectories. Using real-world AIS data trajectories from which motion features were extracted to represent each trajectory information while avoiding the position, focusing the classification on the movement patterns.

It is noteworthy that these previous studies focused on the preparation of real-world data to obtain improvements in the data mining process. In particular, this preparation seeks to solve problems such as trajectory noise or data imbalance, which are considered some of the most important challenges of this type of problems [3] due to the reduction of effectiveness that they cause in the algorithms.

For this proposal, the clustering problem is studied for its application for the same real-world data. The proposal aims to the detection of representative clusters for a trajectory dataset that includes different ship types.

For the detection of anomalies, an element of great interest is the use of context information to improve the solution results. This information represents additional knowledge that can be included in the solution applied to the problem, improving the original results based on new knowledge that differs from the original data.

In the case of the proposed clustering approach this context information can appear in two ways: by means of additional information included to improve the clustering itself (using clustering as its own solution to the anomaly detection problem) or as the clustering output itself, being new information extracted from the clustering and used inside an additional process that allows to improve a data mining solution such as the classifier already studied.

For the maritime environment an interesting context information is the one proposed by [4]. This approach takes advantage of the knowledge provided by the International Maritime Organization (IMO) [5] to adjust the clusters generated with the movement guidelines suggested by said organization. These directives include a series of navigational guidelines recommended to ships when operating in specific areas of the world. However, this information is not usable in any problem, because these recommended routes only exist in specific sites due to their danger or traffic flow. Therefore, a possible solution is to replicate these routes by using clustering algorithms to define common routes within the maritime environment. Therefore, this study proposes an adaptation of a clustering algorithm to trajectory data.

Specifically, in order to adapt the clustering solution to maritime trajectory data, two approaches are proposed and analysed. On the one hand, a technique used to measure the similarity of time series (and consequently of trajectories) is studied. Specifically, the DTW algorithm is used due to its proven efficiency, although the solution is applicable with different trajectory similarity metrics.

On the other hand, prior knowledge is used to extract a set of kinematic characteristics of each trajectory, forming a single multi-dimensional instance (one for each feature) that summarizes the whole trajectory.

Both techniques are applied on the chosen clustering algorithm, DBSCAN, obtaining a series of clusters that provide information about the context in which the maritime data operates. This information has been found to be useful for anomaly detection problems to be applied within this dataset.

This paper is organized as follows: In section II the state-of-art methods in data mining data preparation and clustering on trajectories are analyzed. In section III the system implemented to obtain the clustering applied to trajectories is presented, while in IV the results of the proposed clustering approaches are analyzed. Finally, the conclusions and future work perspectives are presented in the section V.

II. State of the Art

A. Data preprocessing and data imbalance

The real-world data used in the problem requires preparation that allow the classification algorithms to perform more effectively. In particular, the data used comes from the dataset provided by the Danish Maritime Authority [6], and consists in Automatic Identification System (AIS) measurements [7], which is an International Maritime Organisation (IMO) standard mandatory on most maritime vehicles.[8]

This information resource has demonstrated problems [9] that require data preparation for use in data mining processes. Typical cleaning approaches include the removal of inconsistencies, null values and noisy values. The best solution is to analyze the data and the problem to ensure the best possible cleaning process, although the noisy trajectory information could be smoothened by the use of state estimation algorithms.

Between those trajectory estimation problems the Kalman Filter (KF) [10], and variations like the extended version (EKF) [11] are widely used to predict the position according to different motion models. While the adaptive filters allow the adjustment of the predictions to multiple dynamic models. The most prominent example is the Interacting Multiple Model (IMM) filter [12] for its proven effectiveness [13].

The class imbalance is another common problem in the field of data mining as it affects the results in any area and its undeniable presence when using real world data. The problem must be addressed by the use of specific data mining algorithms that consider the imbalance between the used data, although the most common approach is to apply data level algorithm that allow the resampling of instances [14], [15], [16].

Between the data mining problems, most authors prefer to use balanced datasets ignoring the imbalance in the real-world data, although few authors put emphasis on solving this type of problem, one example is [17] which uses the oversampling algorithm SMOTE [18] to create new trajectories of the minority class.

B. Trajectory clustering

In previous studies we used the above-mentioned data preparation within a classification problem, seeking to understand the data itself and to be able to define the ship type from the proposed information. In this study the proposal is the application of the data mining problem from an unsupervised learning perspective, applying clustering techniques on the previously studied data to extract new useful information applicable to the anomaly detection problem.

In the literature there are several approaches for data mining problems applied over trajectory data, being of special interest the way to transform the data coming from a trajectory into data accepted as an input for the data mining technique.

The trajectories are composed of several positions along the time, so the two possible approaches are the transformation of the used algorithm to be applicable for this kind of data [4], [19], [20] or the transformation of the data into a more suitable input accepted by an unmodified algorithm.

For the transformation of the algorithm a common approach is the use of specific metrics that consider the whole trajectory to measure the distance between instances, for example the use of similarity metrics as a distance measure for clustering approaches.[20], [21] There are several applicable distance metrics useful for the trajectory similarity computation [22] although it is important to note that these metrics tend to have a high computational complexity O(mn) since they usually need to operate with all the points of both trajectories. Among the similarity metrics, DTW [23] is a proven robust technique, although it has the disadvantage of being impaired when there is excessive noise in any of the trajectories.

A possible solution to the high complexity problem is the compression of trajectories, which allows to obtain trajectories of a smaller size that in exchange for a loss of information manage to summarize the original trajectory. The most classical algorithm is Douglas-Peucker and an example of its use can be found in [24], which uses this algorithm prior to the clustering problem.

A typical approach to data transformation is the extraction of features that summarize the trajectory into a set of specific values using kinematics and context information. An example of this can be found in [25]–[27] for classification problems or in [28]–[31] applied over clustering problems.

In the literature there are many problems that use clustering approaches applied on trajectories, [4] makes a DBSCANSD clustering that is adapted to work with the speed and distance of the trajectory, also includes in its logical process the information of maritime routes provided by the IMO [5].

Laxhammar presents a study specifically dedicated to the detection of trajectories by applying clustering using a Sequential Hausdorff distance metric. [21], [31]

On the other hand, [20] studies the possibility of using fragments of trajectories to extract more precise information about the clustering to be applied. His key observation being that clustering trajectories as a whole could miss common sub-trajectories of interest for the data mining approach.

With these examples it can be seen that applying clustering on trajectory data is a problem of great interest in different application areas. The result of the obtained clusters is directly applicable for anomaly detection, being possible to extract a representative path from each obtained cluster, being possible the anomalies detection by the study of the representative paths. Thus making an approximation to the routes proposed by the IMO [5] which is applicable to data from any region instead of only to those defined by the IMO.

Another interesting application is the use of clusters for other data mining problems, for example they give the opportunity to perform a more informed class balancing by eliminating trajectories with the knowledge obtained about their similarity, instead of performing an uninformed undersampling.

III. Proposed Architecture

To achieve the proposed data mining problem, a two-step process is performed. First, it is necessary to perform a data preparation and cleaning process in order to improve the results obtained by the techniques to be applied.

Secondly, all the necessary process to perform clustering on the prepared trajectories will be applied.

A. *Data preparation*

The used data[6] consists on AIS measures without any organisation, so the process needs group individual measurements into trajectories useful for the process, cleaning up noisy or erroneous measurements to improve data quality.

To smooth the noisy measurements as a whole, a tracking filter can be applied on each trajectory. The IMM is a well-proven filter that makes it possible to smooth the measurements by combining the different motion modes.

AIS data contains a lot of information, and in the cleaning process it is necessary to decide which is useful for the problem to be solved and which should be discarded.

The given solution uses the kinematic data (timestamp, GPS WGS-84 coordinates, the heading and the speed). Meanwhile, the static information such as ship type, ship name, MMSI (Maritime Mobile Service Identity), etc. which is also available. It is not used directly, except for the cleaning performed:

- The MMSI is used to separate the data into trajectories as it allows each ship to be identified individually.
- The ship type is used as an input data limiter, since only data from cargo ships will be used in the subsequent clustering process.

Table 1. Ship types provided by AIS

Anti-pollution	Cargo	Dredging
Fishing	HSC	Pilot
Port tender	Military	Passenger
Law enforcement	Pleasure	Medical
Reserved	Sailing	SAR
Tanker	Towing	Tug

The class limitation is made for two reasons. On the one hand, the algorithms to be used during the clustering process have a lot of computational complexity, which implies that it is not possible to operate with large data sets in a simple way. On the other hand, when using real world data, including several classes can be detrimental due to the imbalance present in the classes. Therefore, it has been decided to approximate the problem for a single class, being necessary to expand the given solution by including new classes in future studies.

The complexity of clustering algorithms implies the use of a reduced data set, so in this cleaning process it is also necessary to limit the size of the trajectories to be used as well as their quantity.

B. *Clustering algorithm*

In this section the details of the proposed phases for the clustering algorithm selected to solve the data mining problem is introduced, on the other hand, the transformations necessary to convert the input data into a viable input for the algorithm are proposed on the specific subsections.

For the use of clustering applied to trajectory data, one of the most commonly used techniques is the DBSCAN (Density-Based Clustering based on connected regions with high density) algorithm [32].

Algorithm 1 DBSCAN

```
DBSCAN(D,EPS,MinTrs):
    C=0
    for each unvisited point P in dataset D
        mark P as visited
        NeighborPts =regionQuery(P,EPS)
        if sizeof(NeighborPts)<MinTrs
            Mark P as NOISE
        else
            C=next cluster
            expandCluster (…)

expandCluster (P,NeighborPts,C,EPS,MinTrs):
    add P to cluster C
    for each point P’ in  NeighborPts
        if P’ is not visited
            mark P’ as visited
            NeighborPts’ =regionQuery(P,EPS)
            if sizeof(NeighborPts)>=MinTrs
                NeighborPts =
                    NeighborPts joined with NeighborPts’
        if P’ not belongs to any cluster
            add P’ to cluster C

regionQuery(P,EPS):
    return all points within P EPS neighbourhood
```

To work with this algorithm the most important element is the distance metric to be used, since it allows to assign cluster to each instance of the dataset. For this study it will be necessary to define a metric that provides the distance between two trajectories.

Besides this distance metric to be used, the DBSCAN algorithm includes two parameters of interest that need to be defined for the correct performance of the algorithm:

- EPS, which defines the area of the neighbourhood, which in a simplified definition can be interpreted as the distance needed to include elements within the same cluster.
- MinTrs, which defines the density limit of the neighbourhood. A core object is defined only if, the object´s E-neighborhood contains at least MinTrs objects.

To define the distance between the trajectories, the first thing you need to do is to define what the trajectories will be within the algorithm.

Trajectories are representations of the movement made by an object. This motion can be defined as the variation of the object's position over time, which implies that trajectories are composed of instances with two components, position and time.

Therefore, the trajectory is represented by a series of points p_i (which are composed of a series of coordinates) ordered according to a sampling time *t*:

$$T = \{\langle p_1, t_1\rangle, \langle p_2, t_2\rangle, \dots, \langle p_n, t_n\rangle\} \quad (1)$$

In the case of the used ship trajectories, the position is represented by two dimensions in the horizontal plane (x,y) or by latitude and longitude.

The algorithm must be adapted to introduce this information on trajectories instead of single points. The proposal for this study is to use two different approaches:

- Change the distance metric used by the algorithm so that it no longer works by calculating the difference between two points, but by calculating the difference between two trajectories. Specifically, it is proposed to use Dynamic time warping (DTW) as a measure of similarity between trajectories, thus obtaining a useful distance metric for all the points of the two trajectories.
- Apply feature extraction on each trajectory to obtain an individual instance that summarises all points of the trajectory into a single multidimensional point (one dimension for each feature).

1) Distance metric.

DTW is a technique that allows to find the similarity between two time series, therefore it allows to obtain a useful distance metric for the characteristics of the presented problem, since the trajectories present in it are equated to a time series with a position and an associated time, being the DTW a possible distance metric for the clustering algorithm to be implemented.

The technique consists of calculating the distance between any pair of points on the two paths, including those distances in a matrix and finding the path with the minimum difference in that matrix, in summary it seeks to find the minimization of the cumulative distance over potential paths between two time series elements). [23]

This metric is advantageous since it allows to measure the similarity independently of the existing displacement and orientation between the two trajectories, however it has the difficulty of a quadratic time complexity due to the need to compare the trajectories point to point. This means that the proposed technique has a limitation in the dataset to be used, since its scalability does not allow it to be efficient in datasets with thousands of trajectories.[23], [28]

The metric can be translated as the "minimization of the cumulative distance over potential paths between two time series elements" [28]. This implies that the minimum aggregate between distances of two points of the two trajectories is calculated:

$$DTW(T_1, T_2) = \min \left[\sum dist(p_{T1}, p_{T2}) \right] \quad (2)$$

Algorithm 2 DTW

```
DTW(T1,T2):
    initialize matrix D[length(T1)+1,length(T2)+1]
    for i=1 to length(T1)
        D[i,0]=inf
    for i=1 to length(T2)
        D[0,i]=inf
    for each point Pi in T1
        for each point Pj in T2
            Calculate distance between P and P'
            D[Pi, Pj]=distance +
                min (D[Pi-1, Pj], D[Pi,Pj-1], D[Pi-1,Pj-1])
    Return D[length(T1), length(T2)]
```

2) Trajectory features

The use of features to represent the trajectories is a common solution in data mining problems, some examples for the clustering problem can be found in [29]–[31]. This approach allows the summarization of all the trajectory into a simpler set of values.

In previous proposed classification algorithm [2], a large set of features was extracted and tested to assess what information was most relevant to the proposed data mining problem, evaluating which information was most relevant to the proposed data mining problem. The features tested sought to model the ship's behaviour while ignoring the position, with the objective of avoiding the operating environment, so only the kinematic information was considered. These previous studies conclude that the data mining problem can be achieved using only the following three kinematic values:

- The speed module of each measure.
- The course variation between each pair of measures, which is computed through the heading value.
- The trajectory total time, considering the time difference between the first and last measures.

On these kinematic characteristics it is interesting to include statistical measures, such as the average or the maximum, that allow to summarize the whole trajectory.

In comparison to the clusters obtained using the distance metric, it is proposed to use a parallel implementation that considers these features when applying the clustering algorithm. Applying the Euclidean distance metric between the features f_i extracted from the different trajectories.

$$d_E(T_1, T_2) = \sqrt{\sum (f_1 - f_2)^2} \tag{3}$$

IV. RESULTS ANALYSIS

For both approaches to the clustering process, it is necessary to define the EPS and MinTrs parameters to be used. In order to adjust the clusters obtained by the algorithm to the best possible value. A solution used for this adjustment is the calculation of the knee/elbow point [33] on the curve formed by the ordered distances.

Figures Fig. 1 and Fig. 3 show the curves obtained by this procedure for DTW distance and features distance, marking the distance value to be used as EPS parameter. Note that the distances have been normalized prior to the application of the algorithm in order to make it be comparable.

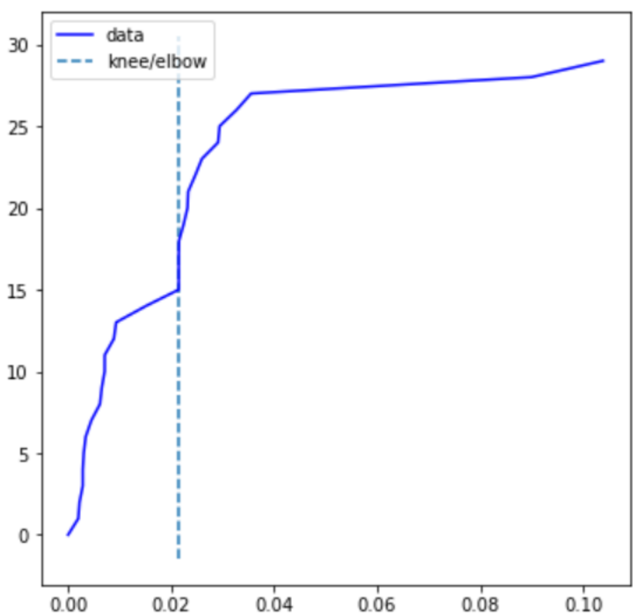

Fig. 1. Knee point for DTW distance metric

With the EPS parameter selected, the clustering algorithm can be applied with each of the variations. Fig. 2 shows the results obtained for the approximation using the DTW distances. A total of 4 clusters (red, green, blue and yellow) are obtained along a noisy trajectory (black) which not assigned to any of them. These clusters stand out for conforming more to the general shape realized by the trajectory, for example the green group is composed of clusters that realize a closed "C" shape while the blue cluster group has more of an "L" shape.

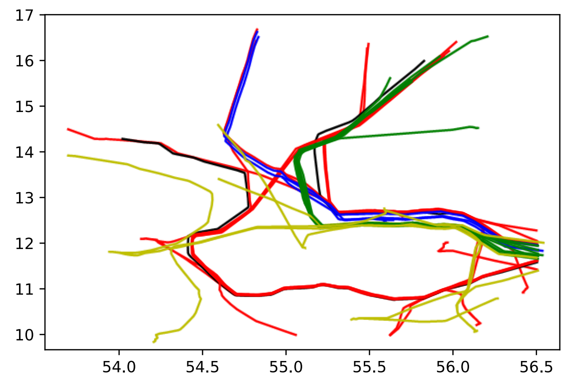

Fig. 2. DTW distances clusters

This DTW approach has the counterpart of being more computationally demanding than the use of features, this is due to the fact that the calculation of features only needs a O(n) complexity as it only need to traverse the points of the trajectory once to calculate the statistical values of each feature. Meanwhile, DTW needs to calculate the distance of all the points of a trajectory, with all the points of another trajectory obtaining an O(nm) complexity.

Considering that the selected clustering algorithm needs the distances of all the trajectories with all the trajectories, the computational complexity of the problem is an important factor to take into account as it greatly reduces the scalability of the problem.

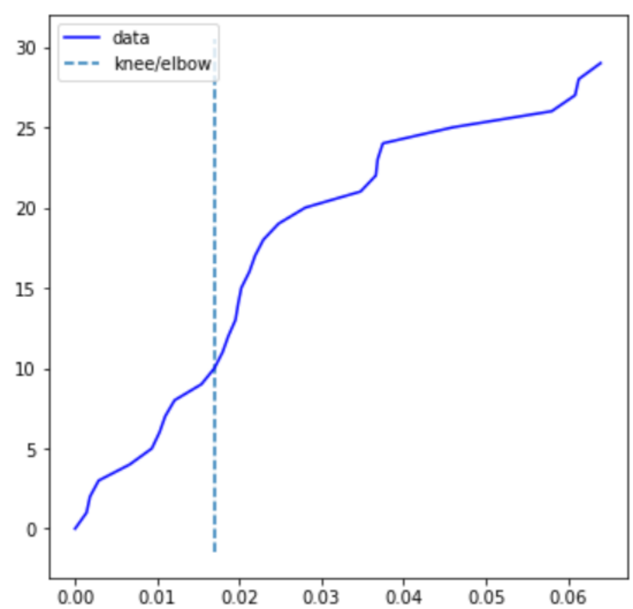

Fig. 3. Knee point for features distance metric

On the other hand, the solution based on the features distance (Fig. 4) is not affected by the shape of the trajectory, there is a large cluster (red) that encompasses most of the trajectories due to the similarity present in them. This is an expected result since the motion of the ships always has similar kinematic characteristics due to their specific motion dynamics.

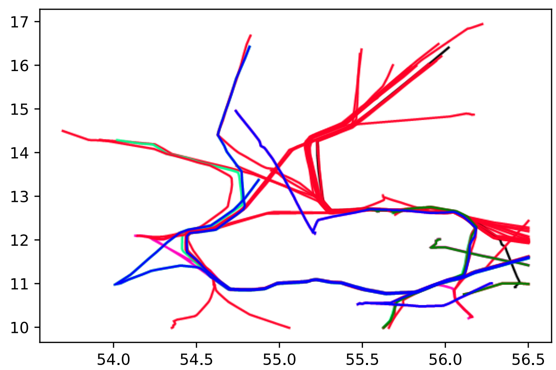

Fig. 4. Feature distance clusters

However, the clusters of short duration trajectories (green) or those with a higher number of tight turns (blue) stand out, as they are directly influenced by the characteristics of time and heading.

This approach has a shorter execution time due to its lower computational complexity, however the clusters created seem to be less informed as there are more apparent differences in the shape of the clusters. Also, using a single set of features to define the entirety of a trajectory can be a negative solution when operating on larger trajectories, since the summarized information is larger which means more difficulty to find specific movement patterns.

V. Conclusions and perspectives

During this study, two viable methods for applying a clustering algorithm on trajectory data have been obtained and analysed. Also, the possible use of these approaches to the data mining problem has been defined. Being useful in anomaly detection problems as a source of additional information or as a data balancing method that can be applied to improve data quality in other data mining problems.

However, the computational complexity of the proposed algorithms demonstrates that it is difficult to scale the solution to a real-world problem where the amount of data is much larger.

The next steps would be to analyze new clustering algorithms or distance approaches that allow better scalability, as well as to apply the solution to a real anomaly detection problem or to improve a data mining algorithm.

VI. Acknowledgement

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

VII. References

[1] D. A. Herrero, D. S. Pedroche, J. G. Herrero, and J. M. M. López, “AIS trajectory classification based on IMM data,” in *2019 22th International Conference on Information Fusion (FUSION)*, Jul. 2019, pp. 1–8.

[2] D. Sánchez Pedroche, D. Amigo, J. García, and J. M. Molina, “Architecture for Trajectory-Based Fishing Ship Classification with AIS Data,” *Sensors*, vol. 20, no. 13, p. 3782, Jul. 2020, doi: 10.3390/s20133782.

[3] B. Krawczyk, “Learning from imbalanced data: open challenges and future directions,” *Prog. Artif. Intell.*, vol. 5, no. 4, pp. 221–232, Nov. 2016, doi: 10.1007/s13748-016-0094-0.

[4] B. Liu, E. N. de Souza, S. Matwin, and M. Sydow, “Knowledge-based clustering of ship trajectories using density-based approach,” in *2014 IEEE International Conference on Big Data (Big Data)*, Washington, DC, USA, Oct. 2014, pp. 603–608. doi: 10.1109/BigData.2014.7004281.

[5] “Ships’ routeing,” IMO. [Online]. Available: https://www.imo.org/en/OurWork/Safety/Pages/ShipsRouteing.aspx

[6] Danish Maritime Authority, “Danish Maritime Authority AIS Dataset.” https://www.dma.dk/SikkerhedTilSoes/Sejladsinformation/AIS/Sider/default.aspx

[7] E. Tu, G. Zhang, L. Rachmawati, E. Rajabally, and G.-B. Huang, “Exploiting AIS data for intelligent maritime navigation: A comprehensive survey,” *IEEE Trans. Intell. Transp. Syst.*, vol. 99, Jun. 2016, doi: 10.1109/TITS.2017.2724551.

[8] IMO, “SOLAS chapter V: Safety of navigation,” Jan. 07, 2002. http://www.imo.org/en/OurWork/facilitation/documents/solas%20v%20on%20safety%20of%20navigation.pdf (accessed May 21, 2020).

[9] A. Harati-Mokhtari, A. Wall, P. Brooks, and J. Wang, “Automatic Identification System (AIS): Data Reliability and Human Error Implications,” *J. Navig.*, vol. 60, no. 3, pp. 373–389, Sep. 2007, doi: 10.1017/S0373463307004298.

[10] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Trans. ASME–Journal Basic Eng.*, vol. 82, no. Series D, pp. 35–45, 1960, doi: 10.1115/1.3662552.

[11] B. A. McElhoe, “An Assessment of the Navigation and Course Corrections for a Manned Flyby of Mars or Venus,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-2, no. 4, pp. 613–623, Jul. 1966, doi: 10.1109/TAES.1966.4501892.

[12] H. A. P. Blom and Y. Bar-Shalom, “The interacting multiple model algorithm for systems with Markovian switching coefficients,” *IEEE Trans. Autom. Control*, vol. 33, no. 8, pp. 780–783, Aug. 1988, doi: 10.1109/9.1299.

[13] X. R. Li and V. P. Jilkov, “Survey of maneuvering target tracking. part v: multiple-model methods,” *IEEE Trans.*

Aerosp. Electron. Syst., vol. 41, no. 4, pp. 1255–1321, Oct. 2005, doi: 10.1109/TAES.2005.1561886.

[14] A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, *Learning from Imbalanced Data Sets*. Cham: Springer International Publishing, 2018. doi: 10.1007/978-3-319-98074-4.

[15] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explor. Newsl.*, vol. 6, no. 1, p. 20, Jun. 2004, doi: 10.1145/1007730.1007735.

[16] A. Gosain and S. Sardana, "Handling class imbalance problem using oversampling techniques: A review," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Udupi, Sep. 2017, pp. 79–85. doi: 10.1109/ICACCI.2017.8125820.

[17] T. Zhang, S. Zhao, and J. Chen, "Research on Ship Classification Based on Trajectory Association," in *Knowledge Science, Engineering and Management*, vol. 11775, C. Douligeris, D. Karagiannis, and D. Apostolou, Eds. Cham: Springer International Publishing, 2019, pp. 327–340. doi: 10.1007/978-3-030-29551-6_28.

[18] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[19] M. Nanni and D. Pedreschi, "Time-focused clustering of trajectories of moving objects," *J. Intell. Inf. Syst.*, vol. 27, no. 3, pp. 267–289, Nov. 2006, doi: 10.1007/s10844-006-9953-7.

[20] J.-G. Lee, J. Han, and K.-Y. Whang, "Trajectory Clustering: A Partition-and-Group Framework," p. 12.

[21] R. Laxhammar and G. Falkman, "Online Learning and Sequential Anomaly Detection in Trajectories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 6, pp. 1158–1173, Jun. 2014, doi: 10.1109/TPAMI.2013.172.

[22] H. Su, S. Liu, B. Zheng, X. Zhou, and K. Zheng, "A survey of trajectory distance measures and performance evaluation," *VLDB J.*, vol. 29, no. 1, pp. 3–32, Jan. 2020, doi: 10.1007/s00778-019-00574-9.

[23] D. J. Berndt and J. Clifford, "Using Dynamic Time Warping to Find Patterns in Time Series," p. 12.

[24] L. Zhao and G. Shi, "A trajectory clustering method based on Douglas-Peucker compression and density for marine traffic pattern recognition," *Ocean Eng.*, vol. 172, pp. 456–467, Jan. 2019, doi: 10.1016/j.oceaneng.2018.12.019.

[25] K. Sheng, Z. Liu, D. Zhou, A. He, and C. Feng, "Research on Ship Classification Based on Trajectory Features," *J. Navig.*, vol. 71, no. 1, pp. 100–116, Jan. 2018, doi: 10.1017/S0373463317000546.

[26] Y. Zhou, W. Daamen, T. Vellinga, and S. P. Hoogendoorn, "Ship classification based on ship behavior clustering from AIS data," *Ocean Eng.*, vol. 175, pp. 176–187, Mar. 2019, doi: 10.1016/j.oceaneng.2019.02.005.

[27] P. Kraus, C. Mohrdieck, and F. Schwenker, "Ship classification based on trajectory data with machine-learning methods," in *2018 19th International Radar Symposium (IRS)*, Bonn, Jun. 2018, pp. 1–10. doi: 10.23919/IRS.2018.8448028.

[28] S. Salvador and P. Chan, "Toward accurate dynamic time warping in linear time and space," *Intell. Data Anal.*, vol. 11, no. 5, pp. 561–580, Oct. 2007, doi: 10.3233/IDA-2007-11508.

[29] D. Fernando Rodriguez and A. Enrique Ortiz, "Detecting Representative Trajectories in Moving Objects Databases from Clusters," in *Information Technology and Systems: Proceedings of ICITS 2020*, vol. 1137, Cham: Springer International Publishing, 2020, pp. 141–151. doi: 10.1007/978-3-030-40690-5.

[30] B. Liu, E. N. de Souza, C. Hilliard, and S. Matwin, "Ship Movement Anomaly Detection Using Specialized Distance Measures," p. 8.

[31] R. Laxhammar, "Anomaly Detection for Sea Surveillance," p. 8.

[32] M. Ester, H.-P. Kriegel, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," p. 6.

[33] A. Starczewski, P. Goetzen, and M. J. Er, "A New Method for Automatic Determining of the DBSCAN Parameters," *J. Artif. Intell. Soft Comput. Res.*, vol. 10, no. 3, pp. 209–221, Jul. 2020, doi: 10.2478/jaiscr-2020-0014.