International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 16, No.1 (Sep-24)

https://dx.doi.org/10.12785/ijcds/1601107

Ship Movement Analysis Based on Automatic Identification System (AIS) Data Using Convolutional Neural Network and Multiple Thread Processing

Yosia Adriel Kornelius[1], Maybin K. Muyeba[2] and Harco Leslie Hendric Spits Warnars[1]

[1]*Department of Computer Science, BINUS Graduate Program – Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia, 11380*
[2]*School of Science, Engineering and Environment, University of Salford, United Kingdom*

Received 1 Apr. 2024, Revised 27 May 2024, Accepted 6 Jun. 2024, Published 20 Sep. 2024

Abstract: Automatic Identification System (AIS) data is one of the most common and widely used datasets in the maritime industry. This dataset is a useful source of information regarding maritime traffic for both individuals and businesses. The reliability of this data and the long-distance transmission over the sea are the primary motivating factors behind its utilization. A wide variety of research projects are currently being carried out on this AIS data. Some of the applications that are being investigated include the detection of ship travel anomalies, the monitoring of ship security, the detection of ship collisions, and the pursuit of shipment trajectory tracking. A number of different methods of machine learning and deep learning are also being utilized in order to perform the analysis of the data. Nevertheless, the vast majority of the studies that have been done up to now have been carried out without any analysis of the consequences of concurrent processing of AIS data. This study conducted a ship movement analysis using AIS data.
This study performed investigation and evaluation in order to see the impact of different numbers of threads processing during the analysis of AIS data. The number of threads used corresponds to the number of cores available on the CPU. Deep learning CNN model used for ship movement classification analysis. This study captured the speed, accuracy, and CPU utilization while performing AIS data analysis. The result shows a noticeable reduction of approximately 40% in processing time while the number of threads increased with no impact on accuracy. The study also found that CPU utilization increased with the increase in the number of threads used to do analysis.

Keywords: Automatic Identification System data, AIS data, Convolutional Neural Network, Multithread Processing, Parallel Processing.

1. Introduction

Transportation of goods from one country to another country can be done by land, water, and air. Since not every country is connected by land and sending goods by air is expensive, goods transport is primarily done by sea using ships. Transporting goods by sea has never been as busy as today. Many ships are travelling through the sea making them more vulnerable to collisions. Studying ship transportation is crucial for various reasons. Firstly, it is the backbone of global trade, with approximately 90% of goods being transported through ships [1]. Understanding and predicting the ship transportation data can lead to improved logistics and sustainability practices. Research in this field can contribute to the development of advanced technologies and safety measures for ships, ensuring safer and more efficient transportation of goods across the world. Furthermore, having a better understanding of the financial effects of ship transportation can help with decision-making when it comes to investments and regulation in the marine sector. According to [2] the International Convention for Safety of Life at Sea (SOLAS), it was decided that every ship on international voyages having more than 300 gross tonnage, all cargo ships having more than 500 gross tonnage, and all passenger ships are required to install Automatic identification system (AIS) equipment. Right now, because the technology in the communication network has matured, it is possible to do all-day tracking of ship navigation data. With also the advancement of Big Data [3], Cloud Computing, and Machine Learning [4], the gathering of all ship data has become possible. However, many hidden factors affect the navigational patterns recorded by AIS data, such as the seamanship of officers, ship maneuverability, geography, and rules of collision avoidance. AIS data contains the location of the ship (latitude and longitude), date, heading, ship name, and other common information that can provide concise information about the ship at some particular time. These data are obtained by using sensors that are either onboard the ship or on-shores and transmitted through the sea in almost real-time. Based on [5], [6], AIS data are preferred compared to the old data from radar, sonar, or CCTV (Closed Circuit TeleVision). Below are some of the advantages of AIS data:

E-mail address: yosia.kornelius@binus.ac.id, k.muyeba@salford.ac.uk, shendric@binus.edu

1) It is possible to use AIS data for path planning. When the risk of collision reaches a certain threshold, the path planning component will develop an alternative route that will assure safety while minimizing the impact on the sailing time and distance of the own ship. This will allow the ship to avoid potential collisions. In addition, a thorough analysis of the various path planning strategy and approaches for route planning is also possible with AIS data.
2) Real-time anomaly detection is capable of identifying potential security and navigation safety issues. As a result, it is highly important for an intelligent navigation system on board a vessel and for port authorities.
3) AIS data can be delivered and received over a wide range of distances, ranging from 20 nautical miles for an onboard transceiver to a thousand nautical miles for a satellite receiver. Artificial Intelligence Systems (AIS) are relatively unaffected by environmental elements such as sea conditions and weather conditions. Hence, AIS data plays a crucial role in regulating maritime traffic and preventing collisions. Additionally, AIS data derived from the marine intelligent system has garnered considerable interest from the intelligent transportation system sector.
4) Collision prediction evaluates the likelihood of a collision occurring between the user's ship and other target ships by analyzing their anticipated trajectories. If the trajectories of two ships intersect, there is a high risk of collision and it is possible that a collision may occur. Analyzing and predicting AIS data can help prevent this situation.

With the benefit of AIS data like the above, there is already quite a number of research on this topic. For example, in [7], [8], the AIS data is used to avoid collision risk in the port area, especially in the anchorage area. With this research, can have safer ports for all. Another usage of this is described in [9], [10]. AIS data can be used to predict the route of the ship and can be developed further into automatic ship-route design. This research uses clustering analysis to be able to get the route based on AIS data. From some research, it is also can be seen that the AIS data can be developed further to be able to schedule a route [11], [12]. Some researchers also use AIS data to analyze the situation that could happen in a busy port with much bigger ships such as port congestions [13], near miss detections [14], and even to plot an automatic route [12] based on the port conditions. Some research also has been done to analyze traffic on the sea [15], [16], [17]. Some research also done on anomalies that can happen in the AIS data like ship can turn off and turn on the AIS data, so the data are not continuos [18], [19], [20]. Or the ship broadcast wrong ship type like fishing ship but broadcast the AIS data as leisure ship [19]. Also another anomalies are ship deviate from the normal trajectory, close approach with another ship, and late /early arrival to the port [20]. There is a subset of research that is looking for the best algorithm to predict ship movement and ship trajectory [21], [22]. The research uses RNN (Recurring Neural Network) [23] and clustering methods to classify shipment movement [24], [25] and uses machine learning classification models like K-nearest Neighbourhood (KNN), Support Vector Machine (SVM), and Decision Tree (DT). There are also research on new methods by combining Satellite data and AIS data to check ship classification [26] and new route detection method that considers both total maritime traffic and statistics to calculate ship routes, including route width with kernel density estimation analysis (KDE) [16], [27], [28]. Also new method of a Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) to analyse ship manouver point [28]. Another example of research for the automatic reconstruction of a network reflecting the maritime traffic using AIS data with Cumulatif Sum and genetic algorithm [16] are being used to analyse the AIS data dan to get information from it. Convolutional Neural Network (CNN) is a Deep Learning model that is used widely for image classification and is also widely used to analyze AIS data [29], [30]. However, based on [25], it is found that when compared to the other Machine Learning techniques and Deep Learning techniques, CNN is the best method for analyzing and classifying the correct shipment movements. Although there is already numerous research work on AIS data as mentioned above, there is not much work has been done that investigates the time impact of processing the AIS data. By using multiple processes, it can reduce the time needed to process some task. With the research done in [31], [32], can be seen the impact of using multiple processing to perform CNN model training and analysis. However, this depend on the performance of the CPU processing power and how many CPUs that can be used to analyze the data. This research analyzed the impact of multiple processing to get information on ship movement classification from AIS data. It can be seen further below for the impact of multi-thread processing on data preparation and Deep Learning processing.

2. Methodology

The information that were used in this investigation was obtained from a data repository that is accessible to the public. The first step in the process of analyzing the data from the Automatic Identification System (AIS) is to download the data from the website of the National Oceanic and Atmospheric Administration (NOAA) [33]. The information is saved in a CSV(Comma Separated Value) file, and each day's data is kept in a separate CSV file. The data formats are CSV. All of the information on the ship, including its speed, heading, and position at a particular point in time, is contained within the CSV files. Our study does not make use of all of the data for further analysis. The information in the data that is used for analysis is outlined in Table I, which can be found below.

Figure 1 below presents a comprehensive list of tasks required for AIS data analysis. The AIS data initially obtained from the website. Next, established a database as the main storage for the AIS data. After that, implemented a function to gather all the AIS data related to a single day and convert it into an image. These images used as the input for the Convolutional Neural Network (CNN) model. Also implemented a script to classify the data

TABLE I. AIS Data Details for Analysis

Column Name	Description	Used for analysis
Maritime Mobile Service Identity (MMSI)	The Unique ID of the vessel	Y
BaseDateTime	Time in GMT	Y
LAT	Position in Latitude	Y
LON	Position in Longitude	Y
SOG	Speed over Ground, speed of the ship	Y
COG	Course over Ground, direction of the ship	N
Heading	Heading of the ship	Y
VesselName	Name of the ship	N
IMO	IMO number of the ship	N
CallSign	Call sign for the ship	N
VesselType	Type of the ship	N
Status	Vessel navigation status	N
Length	Vessel length	N
Width	Vessel width	N
Draft	Distance of the ship's keel and the waterline	N
Cargo	Cargo type	N
TransceiverClass	AIS transceiver types	N

based on the speed in addition to the image conversion process. In this context, the categories available are static, normal navigation, and manoeuvring. For the next task, distribute the available picture data into training and validation sets for the CNN model. The subsequent step in the flowchart is to define the sequence of the neuron and create the Convolutional Neural Network (CNN) model, followed by the execution of the training procedure with the provided training data. After the training phase, conducted a validation step to evaluate the effectiveness of the Convolutional Neural Network (CNN) model. Finally, gathered the outcomes and subjected them to analysis to derive a conclusion based on the data.

The current system that was utilized is a TensorFlow-based system that employs a CNN model for the analysis of AIS data. The system is required to execute specific operations that transform the data from the positional format into an image, which served as input for the Deep Learning analysis. Using Deep Learning analysis, can determine whether the data indicates a ship in regular navigation, static/stop, or moving. For Multithread processing, Python and multiprocessing library were used. This library is widely used by numerous programmers that want to have parallel processing in their code. Both parallel processing and Deep learning model (multiprocessing and TensorFlow) are matured enough to be used to analyze AIS data. All the stages including data preprocessing, initial data download, deep learning modeling, training, and verification were done using a personal computer with the following specifications: CPU: Intel i3-10100 CPU, RAM: 16 GB, Storage: 250 GB. The overall system design can be seen in Figure 2 below. Utilized a database as a storage for AIS data. First, the data stored per line of CSV file and converted into rows in the database. The database loading performed using a function called data loader process that can load data faster compared to the database insert method. With a database, the user can easily query data with the required parameters such as time, Maritime Mobile Service Identity (MMSI), etc. To perform data preparation, a program gathered the data from Database based on the MMSI, time, heading and speed data to plot an image of the position of the ship based on the AIS data. The output image used as the input for the CNN model using TensorFlow. The result were a ship movement classification whether it is Static, Normal Navigation, or Maneuvering.

This data preparation process takes longer time if it was done using normal function executions. But to make it more efficient, utilized multiple processes running the functions at the same time, so the whole process can be done faster. Figure 3 below illustrates the data preparation process and the output. The data preparation process was processing data based on the unique MMSI. The MMSI ID only be processed one time eventhough there were multiple thread run. This process plot the ship position data and change it into images as the input for the CNN model. As one thread performed the conversion, There were also another process running to do the same plotting and conversion but with different MMSI. These multiple processing that can lead to the reduction of total time required. The number of parallel processes that are available were same number as the available cores in the CPU.

The data preparation process produced image shown in figure 4 below. This figure contains data from one ship in terms of 1 day. The timeframe for the image can be changed bigger or smaller as needed, but for this study, use one full day data in order to see the difference in the time taken to do data preparation and CNN model analysis.

This study performed ship movement analysis on AIS data and saw the impact of parallel processing on the effectiveness of the whole process. Secondly, the study also noticed the impact of parallel processing on the result. There are several examples of this from previous studies [29], [25], [24]. These previous studies were performed to get the classification of ship movement for the data within one day for each unique ship. The

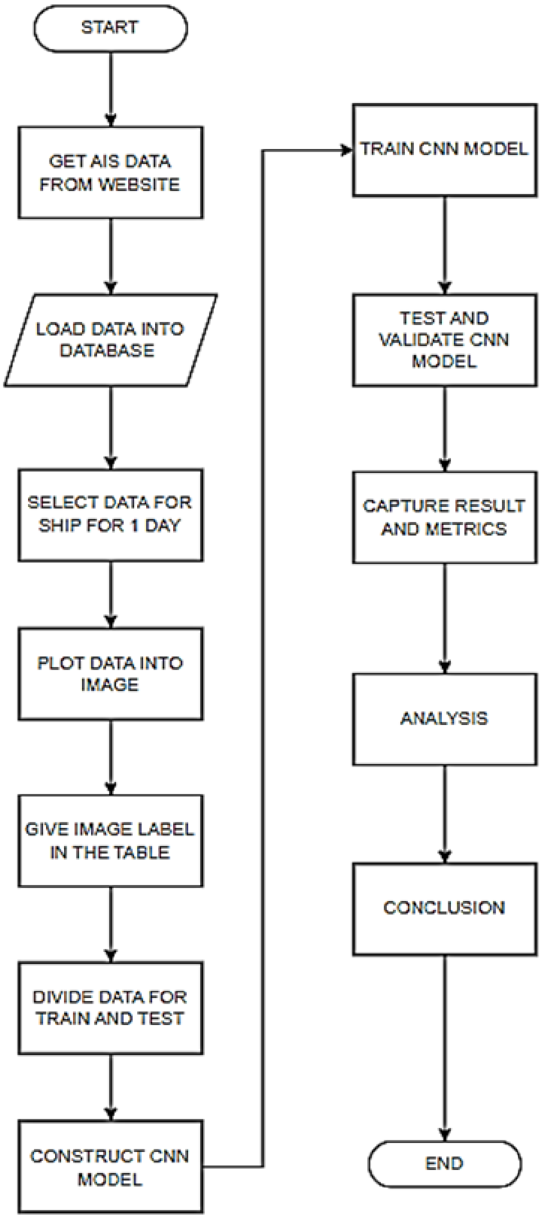

Figure 1. Flowchart of the process for this study

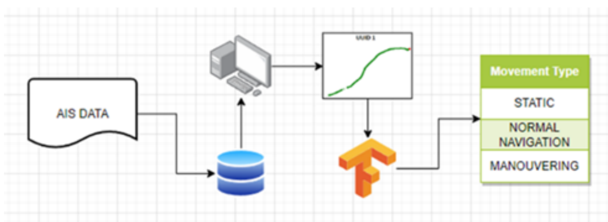

Figure 2. System designed for this study

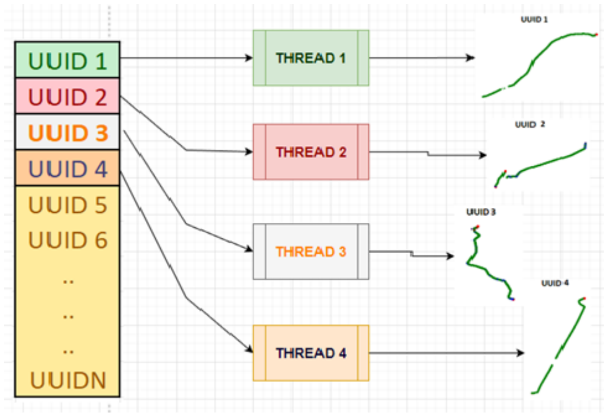

Figure 3. Illustration of the Multithread processing

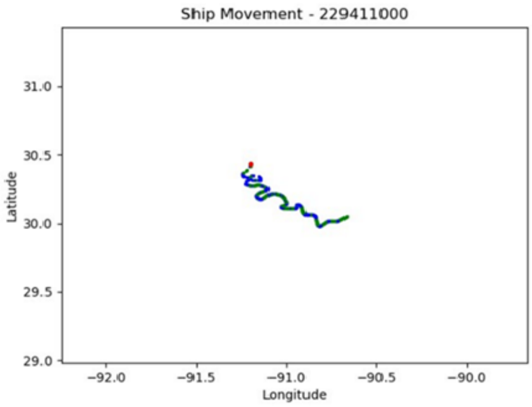

Figure 4. Sample image that created by plotting ship position created for this study with data preparation process

research done in [29] was done using the CNN method, whereas the research done in [25] was done using various Machine Learning methods such as KNN, Clustering, etc and various Deep Learning models including CNN. From the research done by [29], the research found that CNN have Average Accuracy for 98.72%, while using other method like SVM only resulted in 91.73% Accuracy. It is the same result as [25], the research did some comparison between their own CNN model and multiple Machine Learning method that resulted CNN model can reach 92.35% Average Accuracy, while other Machine Learning model like KNN only resulted in 70.23% Average Accuracy. From the research done by the three research works above, it can be seen that CNN is a better method to perform ship movement analysis. The work in [25] used more data compared to others . Our study uses a similar approach as in [25] and with the improvement in the effectiveness and practicality in terms of data filter and data load. Multithread processing was used to make the data processing faster and use a database to easily load and filter the data based on the parameter needed. This paper utilizes the proven CNN model from [25] to analyze ship movements. Based on the same methodology, it was established that CNN also provided the best accuracy compared to other methods. This CNN model has multiple layers that are made up of Convolution Layers, Max-Pooling Layers, a Flatten Layer, and a Dense Layer. The convolution layer does a dot product between two matrices: the confined area of the receptive field is one matrix, and the other matrix is the set of learnable parameters, also referred to as a kernel. This parameter traversed along all the data thus the convolution method. The kernel creates the image representation of that receptive region by sliding across the image's height and breadth during the forward pass. This results in the creation of an activation map, a two-dimensional representation of the image. The pooling layer is responsible for substituting the network's output at specific locations by performing some value passing using the filter like for example for each 2x2. The Max Pool layer take the maximum value for each filter and passing it to the next layer. The other method being used is Batch normalization. It is a method utilized in the training of deep neural networks, which involves standardizing the inputs to a layer for every layer. This batch normalization process leads to the stabilization of the learning process and results in a significant reduction in the number of training epochs necessary for training deep networks. Flatten layer is a neuron in the CNN model that used to simplify the axis of the vector from multiple value to become 1 single value only. This is useful if there is only 1 value needed to determine the classification based on the percentage of probability of this 1 axis value. Flatten id used at the last part of the CNN sequential model. The output of this flatten layer was a 1 dimensional value. The output from the convolutional layers is used as the basis for the classification performed by the Dense Layer. Neurons are included in each layer of a neural network. Another layer that being used is a dropout layer. This dropout layer randomly drop a value based on the parameter passed to the function. With this dropout layer, the data always randomly selected at certain point on the layer and this can help to avoid overfitting on the CNN model. The sequence of the layer for the CNN model and the position of the layer are shown in the figure 5 below. The box representing the same function in Tensorflow library. This same model was used throughout all the experiment.

This study tested two different parts which are the data preparation and CNN model analysis. The 1st part of the test was performed using Python code and using the multiprocess library of Python. For the CNN model, the analysis was performed using the TensorFlow model. The test result captures the time taken for data preparation (change data from position data into image data) and the time taken to do analysis using the CNN model. Besides the time, this study also captures the performance metrics such as Average Precision, Average Recall, and F1 score. The equation for Average Precision, Average Recall, and F1 score given below. The research about this shipment trajectory were evaluated quantitatively. To measure the success of this research, evaluated these 3 parameters from the experiment:

1) Average precision: This is to measure the exactness of predictions with respect to the labels.
2) Average recall: This is to measure how well a system does prediction compared to the total number of correct items
3) F1 Score – which represents the effectiveness of the classifier in identifying positive class.

Table II below shows the structure of a Confusion Matrix. From table II it can be seen how to get the number of True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). TP is the result that the model corectly predict the true class. TN is the result that the model correclty predict the other class. FP is the result of the model predicted as other class, but actually belong to the initial class. While FN is where the model predicted as initial class, but actually it is belong to the other class. The value from these confusion matrix then were used for Average Precision, Average Recall, and F1 Score. The formula for the Average Precision, Average Recall and F1 Score are described in the formula below.

$$\text{Average Precision} = \frac{\sum_{i=1}^{k} \frac{TPi}{TPi+FPi}}{k} \tag{1}$$

Where: TP = True Positive

FP = False Positive

k = Total number of data

$$\text{Average Recall} = \frac{\sum_{i=1}^{k} \frac{TPi}{TPi+FNi}}{k} \tag{2}$$

Where: TP = True Positive

FN = False Negative

k = Total number of data

$$\text{F1 Score} = \frac{2 * AveragePrecision * AverageRecall}{AveragePrecision + AverageRecall} \tag{3}$$

3. Results and Discussions

For the experiment, there were two parts of the whole process from where can see the impact of parallel processing: (1) Data Preparation (converting AIS data to image) and (2) Deep Learning (CNN) processing. Besides that, this study also see the impact of parallel processing on the CPU core utilization. The impact of the parallel processing can be seen in the time taken to complete the data preparation and the CNN training and validation. Furthermore, it have also calculated three performance metrics during the CNN training and validation to see how the impact of parallel processing on the result namely the Average Precision, Average Recall, and F1 Score. For the processing time, can be seen that as the number of parallel threads increased, the time taken to do data preparation and CNN model training and validation is decreased. The result can be found in the table III below. With no parallelism, the whole process runs very slowly and is only finished after 2.5 days. While the analysis done with the 4 thread processing, can see the reduction of the time taken into almost 1 day only.

For the performance metrics, it can be seen that the parallelism did not have any impact on the performance metrics obtained. This can be seen in the result from the test that have been done. The result is available in the table IV below. From all the configuration that have been tested, all have the same value for the Average Precision, Average Call, and F1 Score. As seen from the data preparation column, the time taken to do preparation almost halved for every increase of multi-thread processes performed. From around 10 hours to do the data preparation with no parallelism, now, it can be done in less than 3 hours with 4 parallel processing.

As for the Model Training and Validation duration column, it can be seen that a lot of time was taken to complete this. Using only one process (no parallel processing), took more than two days to complete. While running with two threads processing took one day to complete and four threads processing took less than a day to complete. And for three threads processing took almost the same time compared to four threads processing. The observation on this is that the CPU was fully maximized (constantly on 100%) when running four threads on CNN model training. While running three threads for the same task, it utilizes 94% of the CPU. The whole test was performed using a 4-core CPU. From the study, for the task that is repetitive (i.e. transforming latitude and longitude position into a plot of position of the ship and finally creating image),the time taken to process the data can be reduced by using multiple parallel processing. From here it can be seen that the difference in the time

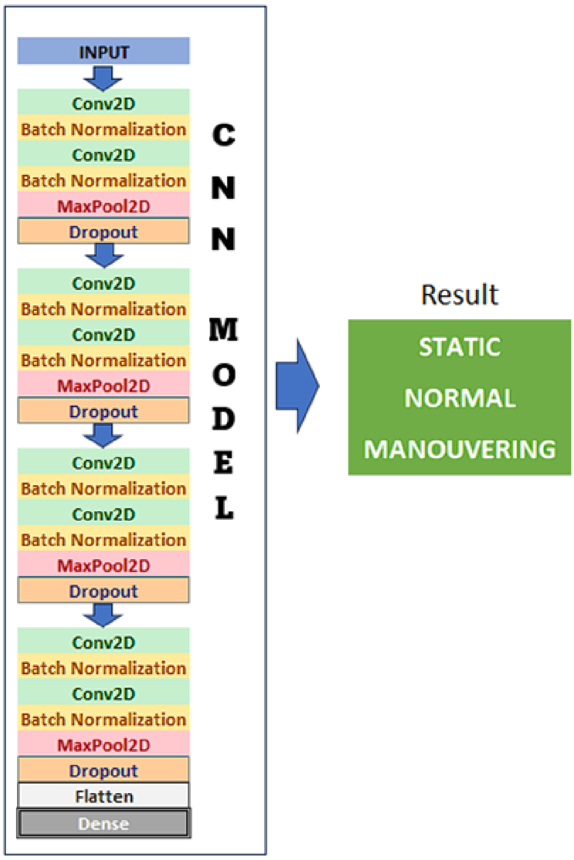

Figure 5. CNN model sequence to do ship movement classification

TABLE II. Confusion Matrix

Movement Modes	Predicted as class i	Predicted as other classes
Class i	True Positive (TP)	False Negative (FN)
Other class	False Positive (FP)	True Negative (TN)

TABLE III. Time Taken to do AIS Data Analysis

Configuration	Data Preparation duration	Model Training and Validation duration	Total time taken
No Parallelism	10.23 Hours	50 Hours	60.23 Hours
2 Thread Processing	6.1 Hours	31.49 Hours	37.59 Hours
3 Thread Processing	4.02 Hours	21.96 Hours	25.98 Hours
4 Thread Processing	2.78 Hours	21.5 Hours	23.28 Hours

TABLE IV. Performance Matrix

Configuration	Average Precision	Average Recall	F1 Score
CNN – No Parallelism	0.6974	0.8229	0.7550
CNN – 2 Thread Processing	0.6974	0.8229	0.7550
CNN – 3 Thread Processing	0.6974	0.8229	0.7550
CNN – 4 Thread Processing	0.6974	0.8229	0.7550

TABLE V. Average Cpu Utilization

Configuration	Average CPU%
CNN – No Parallelism	21%
CNN – 2 Thread Processing	60%
CNN – 3 Thread Processing	94%
CNN – 4 Thread Processing	100%

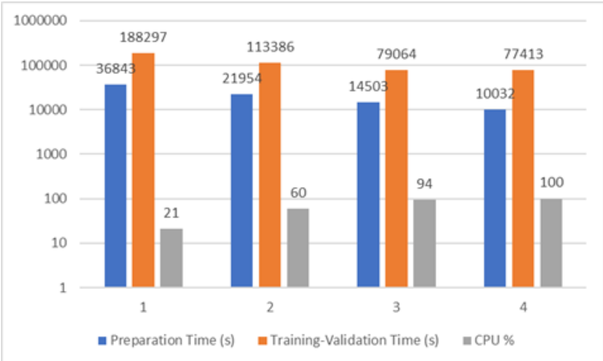

Figure 6. Consolidated data from this study

taken between 3 and 4-core CPUs is not much in terms of processing time. But in terms of utilization, it is always 100% while running using 4 parallel thread processing. It shows that a 4-core CPU was not able to handle the whole process of four parallel thread processing because there is another operating system process that running already, so it is best to have a maximum parallel process to be at least one less than the total number of CPU cores. The result for CPU utilization can be seen in the table below.

In order to consolidate all the results in a single place, all the results have been compiled in Figure 6 below. With this consolidated data, it shows the time taken for Data preparation, CNN training and validation, and CPU percentage. It is concluded that the time taken to complete the process is going down if utilizing more thread during the execution. This happened for both Data Preparation and CNN processing (training and validation).

However, looking at the CPU percentage, this CPU is always increasing for each additional thread. It can be seen in the table V below. This happens because with more parallel processing (i.e. more multithread processing), the whole process broken into multiple jobs. Multiple job running in parallel can make the task completed faster. As for the CPU percentage, the program utilized more threads caused the CPU percentage to increase accordingly.

4. Conclusions and Future Works

In this research, the CNN architecture has been used and tested to extract ship movements from AIS data and perform deep learning analysis including model training and validation. CNN is being used because it has a higher accuracy compared to classification methods like KNN, Decision Trees, and other machine learning methods. It is established for a system that utilizes a Database it will make addition of new AIS data easier and able to filter AIS data based on certain criteria without needing to read from CSV file again. AIS Data was converted to image data and used as input for the CNN System. From the image, it was classified as normal navigation, static, or maneuvering. It can be seen that using parallel processing results in 40% more efficient time usage. From the experiment, can be seen that the whole processing time reduces from 600 minutes to 167 minutes by using 4 multiple parallel processors compared to no parallel processing. From the result, it can be seen also, by using parallel processing, there is no change in the performance metrics and all the precision percentage. But the best result in terms of both the increase of thread and efficiency were achieved when the system utilized maximum core on the system – 1. It is best if 1 CPU core being left out to handle the other tasks i.e. Operating System task. For future research, it is suggested to also apply other processing improvement methods like Hadoop Map Reduce and use the big data database like Hadoop to store the large data and to do data preparation faster compared to the usual normal slower processes. Also, suggest to have more data in Hadoop to analyze the ship movement that can lead to better result and higher accuracy. From the result, it is concluded that using multiprocessing libraries from python helps to utilize all the core CPU utilisation that is available in the physical CPU. From the cost perspective, it is an efficient usage and can result in cost savings. But in terms of the result and processing time, can be seen that there is not much difference between 3 and 4 parallel processors in reducing the time taken to do data preparation and to do Deep learning model training and validation. Another future improvement would be using a more capable hardware processing unit that is optimized for doing Artificial Intelligence like GPU (Nvidia Graphic Processing Unit). With GPU, there can be up to thousands of parallel processing depend on the capabilities of the GPU. It is also proposed to test using a hardware that is optimized like SoC (System on Chip) that have combination of CPU and GPU like Apple M1 and Apple M2 to do the same data preparation and Deep learning model training and validation to see if the optimized SoC can have better performance compared to the normal general-purpose CPU and GPU.

5. Acknowledgement

I would like to express our sincere gratitude to my supervisor, Prof. Harco Leslie Hendric Spits Warnars, S.Kom, M.T.I, Ph.D, for his valuable guidance and support throughout the research process. His expertise and insights were invaluable in shaping our research and helping us overcome challenges. Appreciation also to the colleagues at Binus University for their helpful feedback and support.

References

[1] M. Christiansen, E. Hellsten, D. Pisinger, D. Sacramento, and C. Vilhelmsen, "Liner shipping network design," *European Journal of Operational Research*, vol. 286, pp. 1–20, 10 2020.

[2] M. Gao and G.-Y. Shi, "Ship-handling behavior pattern recognition using ais sub-trajectory clustering analysis based on the t-sne and spectral clustering algorithms," *Ocean Engineering*, vol. 205, p. 106919, 6 2020.

[3] Z. H. Munim, M. Dushenko, V. J. Jimenez, M. H. Shakil, and M. Imset, "Big data and artificial intelligence in the maritime

industry: a bibliometric review and future research directions," *Maritime Policy & Management*, vol. 47, pp. 577–597, 7 2020.

[4] A. Rawson and M. Brito, "A survey of the opportunities and challenges of supervised machine learning in maritime risk analysis," *Transport Reviews*, vol. 43, pp. 108–130, 1 2023.

[5] E. Tu, G. Zhang, L. Rachmawati, E. Rajabally, and G.-B. Huang, "Exploiting ais data for intelligent maritime navigation: A comprehensive survey from data to methodology," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, pp. 1559–1582, 5 2018.

[6] S. Mao, E. Tu, G. Zhang, L. Rachmawati, E. Rajabally, and G.-B. Huang, *An Automatic Identification System (AIS) Database for Maritime Trajectory Prediction and Data Mining*. Springer International Publishing, 2018, pp. 241–257.

[7] Z. Liu, Z. Wu, and Z. Zheng, "A novel model for identifying the vessel collision risk of anchorage," *Applied Ocean Research*, vol. 98, p. 102130, 5 2020.

[8] S. Vukša, P. Vidan, M. Bukljaš, and S. Pavić, "Research on ship collision probability model based on monte carlo simulation and bi-lstm," *Journal of Marine Science and Engineering*, vol. 10, p. 1124, 8 2022.

[9] D. Zhang, Y. Zhang, and C. Zhang, "Data mining approach for automatic ship-route design for coastal seas using ais trajectory clustering analysis," *Ocean Engineering*, vol. 236, p. 109535, 9 2021.

[10] H. Rong, A. Teixeira, and C. G. Soares, "Data mining approach to shipping route characterization and anomaly detection based on ais data," *Ocean Engineering*, vol. 198, p. 106936, 2 2020.

[11] Z. Yan, Y. Xiao, L. Cheng, R. He, X. Ruan, X. Zhou, M. Li, and R. Bin, "Exploring ais data for intelligent maritime routes extraction," *Applied Ocean Research*, vol. 101, p. 102271, 8 2020.

[12] H. Li and Z. Yang, "Incorporation of ais data-based machine learning into unsupervised route planning for maritime autonomous surface ships," *Transportation Research Part E: Logistics and Transportation Review*, vol. 176, p. 103171, 8 2023.

[13] E. a. Kamal Lakhmas, "Using the ais data to enhance smart port congestion optimizing model subtitle as needed (paper subtitle)," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, pp. 1592–1599, 6 2021.

[14] L. Du, O. A. V. Banda, F. Goerlandt, P. Kujala, and W. Zhang, "Improving near miss detection in maritime traffic in the northern baltic sea from ais data," *Journal of Marine Science and Engineering*, vol. 9, p. 180, 2 2021.

[15] C. V. Ribeiro, A. Paes, and D. de Oliveira, "Ais-based maritime anomaly traffic detection: A review," *Expert Systems with Applications*, vol. 231, p. 120561, 11 2023.

[16] D. Filipiak, K. Wecel, M. Stróżyna, M. Michalak, and W. Abramowicz, "Extracting maritime traffic networks from ais data using evolutionary algorithm," *Business & Information Systems Engineering*, vol. 62, pp. 435–450, 10 2020.

[17] Y. Zhang and W. Li, "Dynamic maritime traffic pattern recognition with online cleaning, compression, partition, and clustering of ais data," *Sensors*, vol. 22, p. 6307, 8 2022.

[18] P. Prasad, V. Vatsal, and R. R. Chowdhury, "Maritime vessel route extraction and automatic information system (ais) spoofing detection," in *2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*. IEEE, 2 2021, pp. 1–11.

[19] Z. Yan, X. Song, H. Zhong, L. Yang, and Y. Wang, "Ship classification and anomaly detection based on spaceborne ais data considering behavior characteristics," *Sensors*, vol. 22, p. 7713, 10 2022.

[20] K. Wolsing, L. Roepert, J. Bauer, and K. Wehrle, "Anomaly detection in maritime ais tracks: A review of recent approaches," *Journal of Marine Science and Engineering*, vol. 10, p. 112, 1 2022.

[21] L. Zhang, Q. Meng, Z. Xiao, and X. Fu, "A novel ship trajectory reconstruction approach using ais data," *Ocean Engineering*, vol. 159, pp. 165–174, 7 2018.

[22] B. Murray and L. P. Perera, "A dual linear autoencoder approach for vessel trajectory prediction using historical ais data," *Ocean Engineering*, vol. 209, p. 107478, 8 2020.

[23] M. Gao, G. Shi, and S. Li, "Online prediction of ship behavior with automatic identification system sensor data using bidirectional long short-term memory recurrent neural network," *Sensors*, vol. 18, p. 4211, 11 2018.

[24] Z. Chen, J. Xue, C. Wu, L. Qin, L. Liu, and X. Cheng, "Classification of vessel motion pattern in inland waterways based on automatic identification system," *Ocean Engineering*, vol. 161, pp. 69–76, 8 2018.

[25] X. Chen, Y. Liu, K. Achuthan, and X. Zhang, "A ship movement classification based on automatic identification system (ais) data using convolutional neural network," *Ocean Engineering*, vol. 218, p. 108182, 12 2020.

[26] M. Rodger and R. Guida, "Classification-aided sar and ais data fusion for space-based maritime surveillance," *Remote Sensing*, vol. 13, p. 104, 12 2020.

[27] J.-S. Lee, H.-T. Lee, and I.-S. Cho, "Maritime traffic route detection framework based on statistical density analysis from ais data using a clustering algorithm," *IEEE Access*, vol. 10, pp. 23 355–23 366, 2022.

[28] S. O. Onyango, S. A. Owiredu, K.-I. Kim, and S.-L. Yoo, "A quasi-intelligent maritime route extraction from ais data," *Sensors*, vol. 22, p. 8639, 11 2022.

[29] T. Guo and L. Xie, "Research on ship trajectory classification based on a deep convolutional neural network," *Journal of Marine Science and Engineering*, vol. 10, p. 568, 4 2022.

[30] T. Yang, X. Wang, and Z. Liu, "Ship type recognition based on ship navigating trajectory and convolutional neural network," *Journal of Marine Science and Engineering*, vol. 10, p. 84, 1 2022.

[31] Y. Su, "A parallel computing and mathematical method optimization of cnn network convolution," *Microprocessors and Microsystems*, vol. 80, p. 103571, 2 2021.

[32] D. Datta, D. Mittal, N. P. Mathew, and J. Sairabanu, "Comparison of performance of parallel computation of cpu cores on cnn model," in *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*. IEEE, 2 2020, pp. 1–8.

[33] N. O. for Coastal Management, "Ais data for 2022," 2023. [Online]. Available: https://coast.noaa.gov/htdata/CMSP/AISDataHandler/2022/index.html

Yosia Adriel Kornelius received the Sarjana Teknik (equivalent to B.Eng) degree in electrical engineering from the Institut Teknologi Bandung. He is now working as a Team Lead Production Support in banking industry. He has interest in Deep Learning, Parallel Processing, Big Data and Cloud computing. He have some professional certification in Cloud Computing. He can be contacted at email: yosia.kornelius@binus.ac.id.

Dr Maybin K. Muyeba received a BSc Maths degree in 1989, an MSc degree in Computer Science in 1992 and a PhD degree in Data Mining in 2000 from The University of Manchester, UK. He has worked as a Senior lecturer and researcher in Data Science for over 10 years. He is a director of a public and private sector consulting firm in machine learning and data analytics, and a teaching Fellow in Data Science at University of Salford. He is contactable at k.muyeba@salford.ac.uk.

Prof. Harco Leslie Hendric Spits Warnars, S.Kom, M.T.I, Ph.D He is Professor in Computing and Head of Information Systems concentration at department Doctor of Computer Science Bina Nusantara University (https://dcs.binus.ac.id). He is working on some project research with his doctoral Computer Science students in the research area of Computer Science such as Intelligent Information Systems, Intelligent Tutoring Systems, Technology for the disabled, Games, Artificial Intelligence implementation including Data Mining, Machine Learning, and Decision Support Systems. He can be contacted at email: spits.shendric@binus.ac.id.