International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. #, No.# (Mon-20..)

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], 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] *MIST Consulting ltd, Manchester, United Kingdom*

E-mail address: Yosia.kornelius@binus.ac.id, muyebamk@gmail.com, shendric@binus.edu

Received ## Mon. 20##, Revised ## Mon. 20##, Accepted ## Mon. 20##, Published ## Mon. 20##

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 into the consequences of concurrent processing of AIS data. The purpose of this study is to investigate and evaluate the impact that different numbers of thread processing have on the accuracy as well as the processing time. For the analysis of ship movement classification, the deep learning CNN model will be utilized. This study will check the speed, accuracy, and CPU utilization while performing AIS data 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. 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 [1] 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 [2], Cloud Computing, and Machine Learning [3], 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

E-mail:author's email

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 [4] [5], AIS data are preferred compared to the old data from radar, sonar, or CCTV. Below are some of the advantages of AIS data:

1. AIS data can be transmitted to and received from a very long distance (from 20 nautical miles for an onboard transceiver to a thousand nautical miles for a satellite receiver); AIS is less influenced by external factors such as sea conditions and weather conditions. Therefore AIS data plays a significant role in maritime traffic regulation and collision avoidance and AIS data that are based on the marine intelligent system has also drawn much attention from the intelligent transportation system sector.
2. Real-time anomaly detection can identify potential security and navigation safety hazards and therefore is valuable for an onboard intelligent navigation system and for port authorities.
3. Collision prediction assesses the collision risk between own ship and other target ships based on the predicted trajectories. If two trajectories have an intercross, the collision risk is very large and a collision may happen. AIS data can be analyzed and predicted to avoid this.
4. If the collision risk is beyond a certain threshold, the path planning component plans an alternative safe route with minimal cost regarding the sailing time and distance for own ship to avoid potential collision. So a survey of various path-planning methods is also presented.

With the benefit of AIS data like the above, there is already quite a number of research on this topic. For example, in [6], [7], the AIS data is used to avoid collision risk in the port area, especially in the anchorage area. With this research, we can have safer ports for all. Another usage of this is described in [8], [9]. 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. We can also see that the AIS data can be developed further to be able to schedule a route [10], [11].

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 [12], near miss detections [13], and even to plot an automatic route [11] based on the port conditions. Some research also has been done to analyze traffic on the sea [14], [15], [16]. 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 [17], [18], [19]. Or the ship broadcast wrong ship type like fishing ship but broadcast the AIS data as leisure ship [18]. Also another anomalies are ship deviate from the normal trajectory, close approach with another ship, and late /early arrival to the port [19].

There is a subset of research that is looking for the best algorithm to predict ship movement and ship trajectory [20], [21]. The research uses RNN [22] and clustering methods to classify shipment movement [23], [24] 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 [25] 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) [15], [26], [27]. Also new method of a Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) to analyse ship manouver point [27]. Another example of research for the automatic reconstruction of a network reflecting the maritime traffic using AIS data with Cumulatif Sum and genetic algorithm [15] 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 [28], [29]. However, based on [24], 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. We know that by using multiple processes, we can reduce the time needed to process. With the research done in [30], [31] we can see the impact of using multiple processing to perform CNN model training and analysis. However, this will depend on the performance of the CPU processing power and how many CPUs we can use to analyze the data. This research intended to analyze the impact of multiple processing to get information on ship movement classification from AIS data. We can see below the impact of multi-thread processing on data preparation and Deep Learning processing.

2. METHODOLOGY

The information that we 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) [32]. The information is saved in a CSV 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 1, which can be found below.

TABLE 1 AIS DATA THAT IS AVAILABLE AND USED FOR THE ANALYSIS

Column Name	Used for analysis
Maritime Mobile Service Identity (MMSI)	Y
BaseDateTime	Y
LAT	Y
LON	Y
SOG	Y
COG	N
Heading	Y
VesselName	N
IMO	N
CallSign	N
VesselType	N
Status	N
Length	N
Width	N
Draft	N
Cargo	N

Figure 1 below illustrates a detailed sequence of tasks required to examine AIS data. Initially, we retrieve the data from the website. Then we create a database as the designated repository for the AIS data. A procedure will then be carried out to gather all the data for a single day and convert it into an image, which will serve as the input for the CNN model. Along with this image conversion, there will be an automated processes run to categorize the data. The categories are Static, Normal Navigation, and Maneuvering. The next step involves partitioning the image data into training data and validation data. These data will be used as the input for the CNN model. After that the CNN model will be created and the training process of the CNN model will be run using the training data. After the training process, we perform a validation step to measure the performance of the CNN model created. Lastly, we collect the outcomes and perform an analysis of the result and provide a conclusion from the analysis of the data.

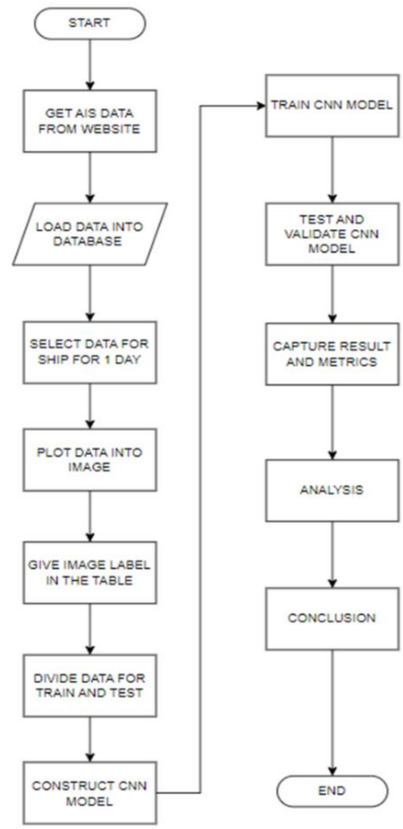

Figure 1. Flowchart of the process for this study

The current system that is being 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 will serve as input for the Deep Learning analysis. Using Deep Learning analysis, we can determine whether the data indicates a ship in regular navigation, static/stop, or moving. The python library that is used for 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 will be 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. We will use a database as a storage for AIS data. First, the data will be stored per line of CSV file and converted into rows in the database. The database loading will utilize a data loader process so it will be faster compared to the database insert method. With a database, we can easily query data with the required parameters that we already load into the database as table 1 above such as time, Maritime Mobile Service Identity (MMSI), etc. After that, a program will gather the data from Database based on the MMSI and certain data to plot an image of the position of the ship based on the AIS data. After that, this image will be used as the input for the CNN model using TensorFlow. The result will be a ship movement classification whether it is Static, Normal Navigation, or Maneuvering.

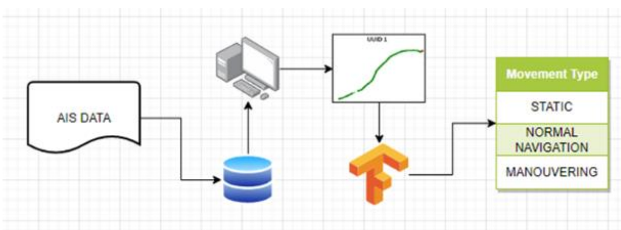

Figure 2. System designed for this study

The first process is to take data from the database and plot the data into a graphic (image). This data preparation process will take longer time if we just use normal function executions. But to make it more efficient, we will utilize multiple processes running the functions at the same time, so this process can be done faster.

The whole study will use Python as the programming language. For the multithread processing, we are utilizing the multiprocess library from Python to do this. Figure 3 below illustrates the multithread process and the output.

The multithread process will process data based on the unique MMSI. This MMSI will only be processed once only in one process. The process will convert the ship position data and convert it into images as the input for the CNN model. There will be a 2nd process running to do the same process that can lead to the reduction of time required. The number of parallel processes that are available will be according to the number of available cores in the CPU. Because using the different MMSI, hence it will not process any duplicate data.

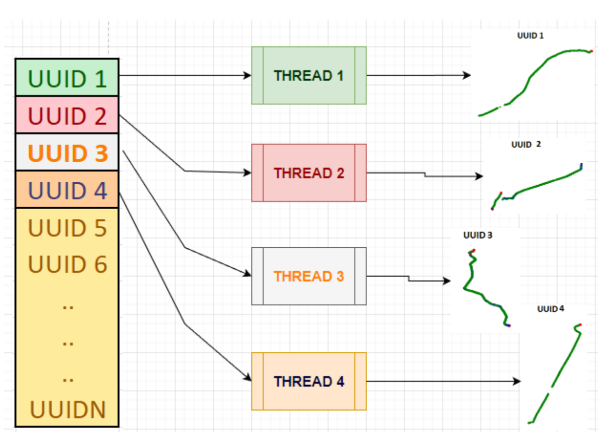

Figure 3. Illustration of the Multithread processing

This process will produce results like in the figure below. This figure contains data from one ship in terms of 1 day. The timeframe for the image can be further changed as needed, but for this study, we use one-day data in order to see the difference in the time taken to do data preparation and CNN model analysis.

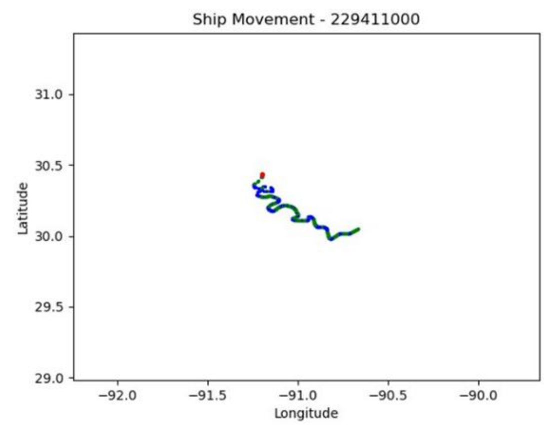

Figure 4. Sample Images created from plotting ship position.

This study aims to analyze AIS data and the impact of parallel processing on the effectiveness of the processes and to determine if there are any impacts on the result. Specifically, the study will analyze the AIS data by performing the ship movement analysis. There are several examples of this from previous research [28], [24], [23]. These studies were performed to get the classification of ship movement for the data within one day for each unique ship. The research done in [28] was done using the CNN method, whereas the research done in [24] was done using various Machine Learning methods such as KNN and various Deep Learning models including CNN. From the research done by [28], 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 [24], 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, we can see that CNN is a better method to perform ship movement analysis. The research done in [24] used more data compared to the other research. Our study uses a similar approach as in [24] and with the improvement in the effectiveness and practicality in terms of data filter and data load. We use multithread processing to make the data processing faster and use a database to easily load and filter the data based on the parameter needed. The CNN model that is being used also taken from the same research to get the best accuracy.

For the CNN model, it will follow the reference model from [24] which from their research is the best model to analyze ship movements that also give the best accuracy. This CNN model will have 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 Convolution process is the core function of the CNN model. This process is what differentiate CNN with other Neural Network model. 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. We are using Max Pool, so it will 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 in CNN model that is used will be put into the last part of the CNN sequential model. The output of this flatten layer will be 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 is being used is a dropout layer. This dropout layer will randomly drop a value based on the parameter passed to the function. With this dropout layer, the data will 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 same model will be used throughout all the experiment.

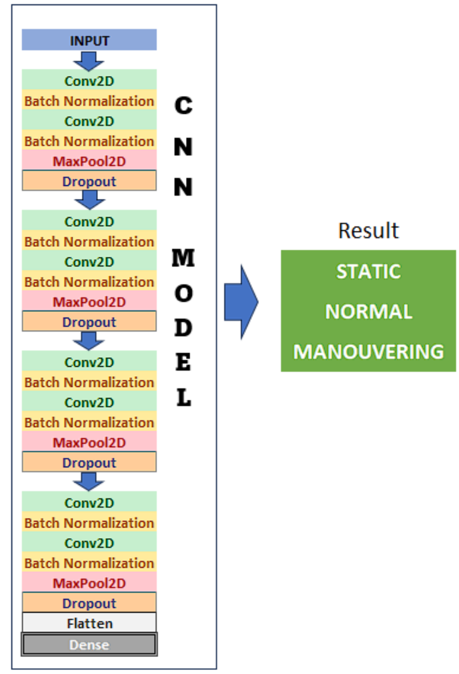

Figure 5. CNN model sequence to do ship movement classification.

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 will be given below.

The research about this shipment trajectory will be evaluated quantitatively. To measure the success of this research, we will evaluate 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 2 below shows the structure of a Confusion Matrix. From table 2 we can see how we can 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 will be 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.

Table 2. 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)

$$Average\_precision = \frac{\sum_{i=1}^{k}(\frac{TPi}{TPi+FPi})}{k} \qquad (1)$$

Where:
TP = True Positive
TN = True Negative
FP = False Positive
FN = False Negative
k = Total number of data

$$Average\_recall = \frac{\sum_{i=1}^{k}(\frac{TPi}{TPi+FNi})}{k} \qquad (2)$$

Where:
TP = True Positive
FN = False Negative
k = Total number of data

$$F1\ Score = \frac{2 * Average\_Precision * Average\_recall}{Average\_Precision + Average\_recall} \qquad (3)$$

To capture the processing time, we capture the time in the beginning of the python code when the program start. After that, the code will run in parallel thread depending on the parameter passed. In this study we will use 1-4 threads as per number of core in the CPU. At the end of the code, another command will capture the time and will calculate the difference of the start time and end time. With this difference, we can see the total time required to perform data preparation or for CNN training and validation.

3. RESULTS AND DISCUSSIONS

For the experiment, there will be two parts of the whole process from where we can see the impact of parallel processing: Data Preparation (converting AIS data to image) and Deep Learning (CNN) processing. Besides that, we 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, we 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, we can see that as we increase the number of parallel threads, the time taken to do data preparation and CNN model training and validation is decreased. The result can be found in the table 2 below. With no parallelism, the whole process runs very slowly and is only finished after 2.5 days. While with the 4 thread processing, we can see the reduction of the time taken into almost 1 day only.

TABLE 2. 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

For the performance metrics, we can see the parallelism did not have any impact on the performance metrics obtained. This can be seen in the result from the test that we have done. We can see the result in the table 3 below. From all the configuration that we have tested, all have the same value for the Average Precision, Average Call, and F1 Score.

TABLE 3. 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

As we can see from the data preparation column, we can see that the time taken to do preparation almost halved for the 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.

As for the training and testing column, we can see a lot of time was taken to complete this process. Using one process, it 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. 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 test was performed using a 4-core CPU.

We can see that 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), we can further down the time taken to process the data by using multiple processing.

From here we can see that the difference in the time 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 we run using 4 parallel thread processing. It shows that a 4-core CPU will not be 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.

TABLE 4. 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%

To see all the results in one place we already put together all the results in Figure 6 below. We can see the time taken for Data preparation, CNN training and validation, and CPU percentage.

We can see that the time taken to complete the process is going down if we have more thread during the execution. This happened for both Data Preparation and CNN processing (training and validation).

However, if we see the CPU percentage, it is increasing for each additional thread.

This happens because with more parallel processing (i.e. more multithread processing), the whole process will be broken into multiple jobs, so it can be completed faster. As for the CPU percentage, the program that utilize more threads will cause the CPU percentage to increase accordingly.

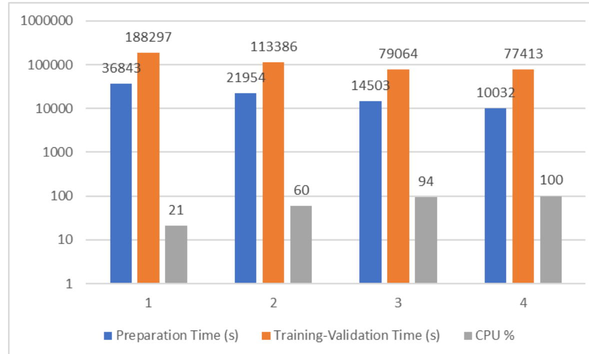

Figure 6. Consolidated data from this study

4. CONCLUSIONS

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 better accuracy compared to the other classification method like KNN, Decision Tree, and other machine learning methods.

We established a system that utilizes Database for easier addition of new AIS data and to filter AIS data based on certain criteria without needing to read from CSV file again. AIS Data converted to image that being used as input for CNN System. From the image, it will be classified as normal navigation, static, or maneuvering.

We can see that using parallel processing can result in 50% more efficient time usage. This can be seen that the whole processing time reduces from 600 minutes to 167 minutes if we are using 4 multiple parallel processing compared to no parallel processing. We can see also from using the parallel processing, there is no change in the performance metrics and all the percentages are the same. But the best result in terms of the increase of thread and increase of efficiency achieved when we are utilizing maximum core on the system – 1. It is best if we leave 1 CPU core to handle the other task i.e. Operating System task.

For future research, we can also applying other task processing improvement method like Hadoop Map Reduce and use the big data database like Hadoop to store the data in bigger size and to do data preparation faster compare to normal process. Also, we suggest to have more data to be used in Hadoop system to analyze the ship movement so we can have better sample that can lead to better result and accuracy.

We can see that using multiprocessing library from python, we can utilize all the CPU core 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, we can see that there is not much difference between 3 and 4 parallel processing in reducing the time taken to do data preparation and to do Deep learning model training and validation.

Another future improvement also we can use 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.

We can also try to use 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.

Acknowledgment

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. We also want to thank our colleagues at **Binus University** for their helpful feedback and support.

References

[1] 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, Jun. 2020, doi: 10.1016/j.oceaneng.2020.106919.

[2] 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, no. 5, pp. 577–597, Jul. 2020, doi: 10.1080/03088839.2020.1788731.

[3] A. Rawson and M. Brito, "A survey of the opportunities and challenges of supervised machine learning in maritime risk analysis," *Transp Rev*, vol. 43, no. 1, pp. 108–130, Jan. 2023, doi: 10.1080/01441647.2022.2036864.

[4] 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, no. 5, pp. 1559–1582, May 2018, doi: 10.1109/TITS.2017.2724551.

[5] 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," 2018, pp. 241–257. doi: 10.1007/978-3-319-57421-9_20.

[6] 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, May 2020, doi: 10.1016/j.apor.2020.102130.

[7] S. Vukša, P. Vidan, M. Bukljaš, and S. Pavić, "Research on Ship Collision Probability Model Based on Monte Carlo Simulation and Bi-LSTM," *J Mar Sci Eng*, vol. 10, no. 8, p. 1124, Aug. 2022, doi: 10.3390/jmse10081124.

[8] H. Rong, A. P. Teixeira, and C. Guedes Soares, "Data mining approach to shipping route characterization and anomaly detection based on AIS data," *Ocean Engineering*, vol. 198, p. 106936, Feb. 2020, doi: 10.1016/j.oceaneng.2020.106936.

[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, Sep. 2021, doi: 10.1016/j.oceaneng.2021.109535.

[10] Z. Yan *et al.*, "Exploring AIS data for intelligent maritime routes extraction," *Applied Ocean Research*, vol. 101, p. 102271, Aug. 2020, doi: 10.1016/j.apor.2020.102271.

[11] H. Li and Z. Yang, "Incorporation of AIS data-based machine learning into unsupervised route planning for maritime autonomous surface ships," *Transp Res E Logist Transp Rev*, vol. 176, p. 103171, Aug. 2023, doi: 10.1016/j.tre.2023.103171.

[12] K. L. Et. al., "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, no. 9, pp. 1592–1599, Jun. 2021, doi: 10.17762/turcomat.v12i9.3552.

[13] L. Du, O. A. Valdez Banda, F. Goerlandt, P. Kujala, and W. Zhang, "Improving Near Miss Detection in Maritime Traffic in the Northern Baltic Sea from AIS Data," *J Mar Sci Eng*, vol. 9, no. 2, p. 180, Feb. 2021, doi: 10.3390/jmse9020180.

[14] C. V. Ribeiro, A. Paes, and D. de Oliveira, "AIS-based maritime anomaly traffic detection: A review," *Expert Syst Appl*, vol. 231, p. 120561, Nov. 2023, doi: 10.1016/j.eswa.2023.120561.

[15] D. Filipiak, K. Węcel, M. Stróżyna, M. Michalak, and W. Abramowicz, "Extracting Maritime Traffic Networks from AIS Data Using Evolutionary Algorithm," *Business & Information Systems Engineering*, vol. 62, no. 5, pp. 435–450, Oct. 2020, doi: 10.1007/s12599-020-00661-0.
[16] Y. Zhang and W. Li, "Dynamic Maritime Traffic Pattern Recognition with Online Cleaning, Compression, Partition, and Clustering of AIS Data," *Sensors*, vol. 22, no. 16, p. 6307, Aug. 2022, doi: 10.3390/s22166307.
[17] P. Prasad, V. Vatsal, and R. Roy 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, Feb. 2021, pp. 1–11. doi: 10.1109/ICAECT49130.2021.9392536.
[18] 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, no. 20, p. 7713, Oct. 2022, doi: 10.3390/s22207713.
[19] K. Wolsing, L. Roepert, J. Bauer, and K. Wehrle, "Anomaly Detection in Maritime AIS Tracks: A Review of Recent Approaches," *J Mar Sci Eng*, vol. 10, no. 1, p. 112, Jan. 2022, doi: 10.3390/jmse10010112.
[20] 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, Jul. 2018, doi: 10.1016/j.oceaneng.2018.03.085.
[21] 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, Aug. 2020, doi: 10.1016/j.oceaneng.2020.107478.
[22] 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, no. 12, p. 4211, Nov. 2018, doi: 10.3390/s18124211.
[23] 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, Aug. 2018, doi: 10.1016/j.oceaneng.2018.04.072.
[24] 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, Dec. 2020, doi: 10.1016/j.oceaneng.2020.108182.
[25] M. Rodger and R. Guida, "Classification-Aided SAR and AIS Data Fusion for Space-Based Maritime Surveillance," *Remote Sens (Basel)*, vol. 13, no. 1, p. 104, Dec. 2020, doi: 10.3390/rs13010104.
[26] 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. 23355–23366, 2022, doi: 10.1109/ACCESS.2022.3154363.
[27] 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, no. 22, p. 8639, Nov. 2022, doi: 10.3390/s22228639.
[28] T. Guo and L. Xie, "Research on Ship Trajectory Classification Based on a Deep Convolutional Neural Network," *J Mar Sci Eng*, vol. 10, no. 5, p. 568, Apr. 2022, doi: 10.3390/jmse10050568.
[29] T. Yang, X. Wang, and Z. Liu, "Ship Type Recognition Based on Ship Navigating Trajectory and Convolutional Neural Network," *J Mar Sci Eng*, vol. 10, no. 1, p. 84, Jan. 2022, doi: 10.3390/jmse10010084.
[30] Y. Su, "A parallel computing and mathematical method optimization of CNN network convolution," *Microprocess Microsyst*, vol. 80, p. 103571, Feb. 2021, doi: 10.1016/j.micpro.2020.103571.
[31] 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, Feb. 2020, pp. 1–8. doi: 10.1109/ic-ETITE47903.2020.142.
[32] NOAA Office for Coastal Management, "AIS Data for 2022." Accessed: Mar. 21, 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 Senior DevOps Engineer in one of the biggest global logistics company in the world. He have 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 1990, 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. He is contactable at muyebamk@gmail.com.

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