REPUBLIC OF TURKEY
ISTANBUL GELISIM UNIVERSITY
INSTITUTE OF GRADUATE STUDIES

Department of Electrical and Electronics Engineering

POSITION DETECTION FOR ARBITRARY-ORIENTED SHIPS IN SATELLITE IMAGERY VIA CONVOLUTIONAL NEURAL NETWORK

Master Thesis

Othman Mohammed Ayoob AYOOB

Supervisor
Asst. Prof. Dr. Sevcan KAHRAMAN

Istanbul – 2023

THESIS INTRODUCTION FORM

Name and Surname : Othman mohammed ayoob AYOOB

Language of the Thesis : English

Name of the Thesis : Position Detection For Arbitrary-Oriented Ships In Satellite Imagery Via Convolutional Neural Network

Institute : Istanbul Gelisim University Institute of Graduate Studies

Department : Electrical and Electronics Engineering

Thesis Type : Master

Date of the Thesis : 19/07/2023

Page Number : 81

Thesis Supervisors : Asst. Prof. Dr. Sevcan KAHRAMAN

Index Terms : Ship Detection, Satellite Images, Deep Learning, Semantic Segmentation, ANN, CNN and YOLO-v5.

Turkish Abstract : Kesinlikle Yönlenen Gemiler İçin Uydu Görüntüsünde Konvolusyonel Sinir Ağlariyla Konum Tespiti

Distribution List :

1. To the Institute of Graduate Studies of Istanbul Gelisim University
2. To the National Thesis Center of YÖK (Higher Education Council)

Signature

Othman Mohammed Ayoob AYOOB

REPUBLIC OF TURKEY
ISTANBUL GELISIM UNIVERSITY
INSTITUTE OF GRADUATE STUDIES

Department of Electrical and Electronics Engineering

POSITION DETECTION FOR ARBITRARY-ORIENTED SHIPS IN SATELLITE IMAGERY VIA CONVOLUTIONAL NEURAL NETWORK

Master Thesis

Othman Mohammed Ayoob AYOOB

Supervisor
Asst. Prof. Dr. Sevcan KAHRAMAN

Istanbul – 2023

DECLARATION

I hereby declare that in the preparation of this thesis, scientific ethical rules have been followed, the works of other persons have been referenced in accordance with the scientific norms if used, there is no falsification in the used data, any part of the thesis has not been submitted to this university or any other university as another thesis.

Othman Mohammed Ayoob AYOOB
19/07/2023

TO ISTANBUL GELISIM UNIVERSITY

THE DIRECTORATE OF GRADUATE EDUCATION INSTITUTE

The thesis study of Othman Mohammed Ayoob AYOOB titled as Position Detection For Arbitrary- Oriented Ships in Satellite Imagery Via Convolutional Neural Network has been accepted as MASTER in the department of Electrical-Electronic Engineering by out jury.

Director ______________________________

Asst. Prof. Dr. Sevcan KAHRAMAN

(Supervisor)

Member ______________________________

Asst. Prof. Dr. Yusuf Gurcan SAHIN

Member ______________________________

Asst. Prof. Dr. Kenan BUYUKATAK

APPROVAL

I approve that the signatures above signatures belong to the aforementioned faculty members.

... / ... / 20..

Prof. Dr. Izzet GUMUS

Director of the Institute

SUMMARY

Combining satellite imagery with computer vision techniques enables us to develop numerous applications ranging from precision agriculture to urban planning. In this study, ship detection is performed from satellite images using artificial neural networks. This study will guide important practices that will improve maritime trade through better monitoring and protection of ships.

In this thesis, ANN, CNN and YOLO-v5, one of the most popular recognition algorithms, semantic segmentation models are used. Experimental studies show that the YOLO-v5 algorithm, which is the most popular approach that can detect ships with high performance, has been obtained the best performance results than ANN and CNN algorithms.

Key Words: Ship Detection, Satellite Images, Semantic Segmentation.

ÖZET

Uydu görüntülerini bilgisayarlı görü teknikleriyle birleştirmek, hassas tarımdan şehir planlamasına uzanan çok sayıda uygulama geliştirmemizi sağlamaktadır. Bu çalışmada, yapay sinir ağları kullanılarak uydu görüntülerinden gemi tespiti gerçekleştirilmiştir. Bu çalışma, gemilerin daha iyi izlenmesi ve korunması yoluyla deniz ticaretini geliştirecek önemli uygulamalara yol gösterecektir.

Bu tezde ANN, CNN ve en popüler tanıma algoritmalarından olan YOLO-v5 semantik bölütleme modelleri kullanılmıştır. Yapılan deneysel çalışmalar sonunda gemileri tespit etmeyi yüksek başarımla gerçekleştirebilen en populer yaklaşım olan YOLO-v5 algoritması elde edilmiştir. YOLO-v5 algoritması, ANN ve CNN algoritmalarından daha yüksek performans sağlamıştır.

Anahtar kelimeler: Gemi Tespiti, Uydu Görüntüleri, Semantik Segmentasyon.

TABLE OF CONTENTS

CHAPTER FOUR
RESULT AND DISCUSSION

CONCLUSION AND FUTURE WORKS

ABBREDIVATIONS

AI	**:**	Artificial Intelligence
FSK	**:**	Frequency Shift Keying
IoT	**:**	İnternet of Things
PAN	**:**	Personal Area Network
OFDM	**:**	Orthogonal Frequency Division Multiplexing
TDMA	**:**	Time Division Multiple Access
YOLOv2	**:**	You only look once version 2
CNN	**:**	Convolutional Neural Network
NIR	**:**	Near-infrared
RPN	**:**	region proposal network
SSD	**:**	Single Shot Detector
AP	**:**	Average Precision

LIST OF TABLES

LIST OF GRAPHICS

LIST OF FIGURES

INTRODUCTION

Ship classification and detection systems are used in many areas that may cause problems such as national and local defense, ship traffic control, illegal fishing, piracy, human smuggling and the global trade chain in countries where coastal and strait crossings are intense.

In order to meet such requirements easily, solutions have been developed with many different methods related to ship classification and determination. Research on these methods is heavily divided into two groups. These are the methods used in satellite and SAR images. Although the number of studies on these sources is high, the resulting images are not disclosed to the public (Cordova, Quispe, Inca, Choquehuayta, & Gutierrez, 2020). Also, their resolution is not sufficient for ship classification and detection. As an alternative solution to these methods, Convolutional Neural Network (CNN) with deep learning technology have become very popular in several years. This study focuses on using an optimal deep learning technique by parameters to detect ships from satellite and aerial images and compare optimal models with original deep learning models

The detection of ships in satellite imagery is an important area of research that has practical applications in maritime security and surveillance. Traditional methods of ship detection, such as feature-based approaches, are often limited by the complex and variable nature of the maritime environment. One of the main challenges in ship detection is the high variability and complexity of the maritime environment, which affects the appearance and size of ships in satellite imagery (Alganci, Soydas, & Sertel, 2020). Traditional machine learning techniques, such as support vector machines and random forests, often struggle to handle such variability, which can result in lower detection rates or higher false positive rates (Alghazo, Bashar, Latif, & Zikria, 2021). Research surrounding the detection of ships using satellite optical imagery is extensive regarding the many satellites, algorithmic approaches, and deep learning approaches used (Alibrahim & Ludwig, 2021). This passage discusses the different types of satellites and methods used to detect ships, as well as the challenges in analyzing satellite data. When it comes to identifying ships in satellite imagery, the more general term of "vessel" is not used, as it can also refer to objects other than ships, such as floating docks and canoes. In contrast to optical imagery, which can be affected by clouds and sunlight, synthetic aperture radio (SAR) data is often used for locating ships in aerial images due to its high accuracy (Alzubi1,, Nayyar, & Kumar, 2018) (Ammar, Koubaa, Ahmed, Saad, & Benjdira) (Alibrahim & Ludwig, 2021).

Over the last few decades, a bird's-eye view of the Earth has provided a wealth of information. This is given by means of satellite images, which are particularly helpful in meteorology, planning, checking ecological and climatic change, policing, debacle reaction (Alganci, Soydas, & Sertel, 2020). With the additional progressions in the making of high-goal sensors and high velocity imaging sensors, there has been impressive ascent in both the quality and amount of mistake free and correct satellite photos throughout the past 10 years (Alghazo, Bashar, Latif, & Zikria, 2021).

Object detection from satellite images are very helpful in many areas such as guard and military, surveillance examinations, air terminal observing and evaluating foundation like streets, structures and arenas. Object identification in remote sensing images is an essential task, Although there are several studies and military areas like boat position, marine traffic the executives, and vessel rescue and so on (Bahi & Batouche, 2018) (Bochinski, Senst, & Sikora, 2017).

Object recognition is one of the many purposes of computer vision. Thus, it has been the focal point of examination of (Bahi & Batouche, 2018). Since CNN gained value in object discovery. Deep learning is a class of Artificial Intelligent (AI) strategies that can address information at a few level using various handling layers, considering computerization by means of CNN (Carlet & Abayowa, 2017). CNN have been generally utilized for object location, making another norm for many computer vision and AI applications (Ammar, Koubaa, Ahmed, Saad, & Benjdira). National language processing, image processing, and character recognition are most popular utilizations of CNN. Deep learning can be developed through various ways such as better information, further developed calculations, expanded calculation execution, etc (Alzubi1,, Nayyar, & Kumar, 2018).

Ship detection is an important task in maritime surveillance and security. There are several approaches to detect ships, including using deep learning techniques like Convolutional Neural Networks (CNNs) and You Only Look Once (YOLO) object detection framework. While CNNs are widely used for image classification tasks, YOLO is a popular real-time object detection method that can detect multiple objects in a single frame. Comparing CNNs and YOLO for ship detection can provide insights into their respective strengths and weaknesses in this domain.

Ship detection plays a critical role in various industries, such as global trade, transportation, defense, and fisheries management, as it enables precise monitoring of ships and vessels using object detection techniques on images captured from satellite-based imaging. The ability to accurately detect ships in these images has become increasingly important for multiple purposes, including security, border control, traffic management and environmental monitoring. With the development in technology, ship detection has become an essential tool for ensuring safety, security, and sustainability in various industries (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014).

Satellite imaging-based ship detection has become increasingly important as it allows for wider monitoring of the ocean compared to the earlier methods of coastal radars and shore-based Automatic Identification Systems (AIS) which were limited to near-shore monitoring. With satellite imaging-based ship detection techniques, there is no restriction on the distance from the shore and ships can be precisely monitored from space. This has led to significant improvements in maritime security, border control, traffic management, and fisheries management. By leveraging satellite-based imaging, accurate and reliable detection of ships is possible on a global scale, making it and detection of ships is possible on a global scale, making it an essential tool for ensuring safety, security, and sustainability in various industries (Deng, et al., 2017).

Detecting ships from satellite imagery presents unique challenges compared to other object detection tasks due to the characteristics of the sea surface, including fluctuations in lighting, changing sea conditions, pollution, and variation in ship orientation and size (Domhan, Springenberg, & Hutter, 2015). These factors affect the image clarity and make ship detection more challenging than the detection of other general objects. Hence, ship detection is considered as a unique and challenging case of object detection, and requires advanced algorithms and techniques to achieve accurate and reliable results.

Ship detection from satellite imagery is an important area of research as it has a variety of applications such as in maritime traffic control, fisheries management, and environmental monitoring. The main challenge in ship detection from satellite imagery is the ability to distinguish ships from other objects, such as waves or clouds, and detecting ships accurately in various sea conditions and environments (Claesen & Moor, 2015).

In recent years, several techniques have been developed for ship detection from satellite imagery, including machine learning algorithms, deep learning architectures and traditional image processing methods. Some of the related works in this area include methods for object detection in unmanned aerial vehicle (UAV) surveillance, ship detection in SAR imagery, and ship detection using self-supervised deep learning techniques (Evans, Al-Sahaf, Xue, & Zhang, 2018).

Despite the advances in ship detection techniques, there are still some challenges that need to be addressed, such as the need for robust algorithms that can handle changes in environmental conditions to ability to process large amounts of data efficiently and the need for accurate and reliable detection methods.

The motivation for our work is to explore the potential of using a particular ship detection method in analyzing satellite imagery with high accuracy and reliability. Specifically, it is proposed that the use of a self-supervised deep learning technique to detect ships in Sentinel-2 multi-spectral images. By utilizing this approach, It is aimed to create a robust and accurate ship detection algorithm that can handle various environmental conditions and different types of ships. Ultimately, this study can contribute to improve maritime traffic control, fisheries management and environmental monitoring.

The remainder of this paper is structured as follows. In Section II, we introduce related works including CNN architecture. In Section 3, the proposed YOLO-v5 algorithm has been applied to Sentinel-2 dataset. Experimental results are given in Section 4. Finally, Section 5 concludes the paper

CHAPTER ONE

BACKGROUND

which is a prologue to the issue and portrays the issue explanation, points and targets, and examination questions that are proposed to arrive at the review's objective.

1.1. Ship Detection from Satellite Imagery Introduction

Ship detection from satellite imagery is an important area of research as it has a variety of applications, such as in maritime traffic control, fisheries management, and environmental monitoring. The main challenge in ship detection from satellite imagery is the ability to distinguish ships from other objects, such as waves or clouds, and detecting ships accurately in various sea conditions and environments (Claesen & Moor, 2015).

In recent years, several techniques have been developed for ship detection from satellite imagery, including machine learning algorithms, deep learning architectures, and traditional image processing methods. Some of the related works in this area include methods for object detection in unmanned aerial vehicle (UAV) surveillance, ship detection in SAR imagery, and ship detection using self-supervised deep learning techniques.

Despite the advances in ship detection techniques, there are still some challenges that need to be addressed, such as the need for robust algorithms that can handle changes in environmental conditions, the ability to process large amounts of data efficiently, and the need for accurate and reliable detection methods.

The motivation for our work is to explore the potential of using a particular ship detection method in analyzing satellite imagery with high accuracy and reliability. The use of a self-supervised deep learning technique to detect ships in Sentinel-2 multi-spectral images is proposed. By utilizing this approach, the goal is to create a robust and accurate ship detection algorithm that can handle various environmental conditions and different types of ships. Ultimately, our work can contribute to improved maritime traffic control, fisheries management, and environmental monitoring.

1.2. Literature Review

This segment presents a few central thoughts and wording related with object recognizable proof and satellite images

1.2.1. Satellite image terminology

This section1 characterizes the jargon related with satellite images. The span between two progressive photographs catching a similar locale is alluded to as return to time or transient goal. Factors, for example, a satellite's circle and elevation impact the return to time. As recently expressed, the return to time is basic for the practicality of following and observing boats. The frequency that the satellite's sensor can gather and figure out what is found in the image is alluded to as the range or phantom goal. The frequency of red, green, and blue (RGB) is 0.4 m to 0.7 m. The frequency of close infrared (NIR) is 0.7 m to 1.2 m and is brilliant for perceiving green vegetation. The distance between the focuses of two continuous pixels is indicated by ground examining distance (GSD) or spatial goal. A greater GSD brings about lesser spatial goal and makes it harder to recognize visual subtleties. The GSD is impacted by variables, for example, the satellite's level and the degree of the locale caught by the satellite on the planet. (Called area).

Images containing sets of class marks and jumping confines signifying things the image are a normal information design utilized in object discovery (Liu, Liu, Huo, & Fang, 2022) (Aszemi & Dominic2, 2019). The class name shows what sort of thing is inside the encasing box, which in this investigation is just the class transport. The hopping box insinuates a rectangular district inside the image and is pretty much as little as plausible while yet including the whole thing. The class and jumping boxes of a image are known as the image's ground truth. This information arrangement is utilized to survey models as well as train a few models (Liu, Liu, Huo, & Fang, 2022).

The subsequent part comprises of classifiers prepared with AdaBoost using Haar-like highlights. Haar-like elements are two, three, or four contiguous square shapes with size and area comparative with a reference outline. The square shapes must be positive or negative. Utilizing the essential image, the worth of a Haar-like component is determined by adding the all out of all pixels in the positive square shapes and deducting the pixels in the negative square shapes. Edited images of what is to be distinguished, as well as a few negative examples, are used in the preparation cycle (Li, Fu, Sun, & Sun, 2018), with the trimmed image filling in as the reference outline. Without diving into an excess of detail, AdaBoost makes a classifier by choosing various Haar-like highlights and a limit. The objective is to cultivate a classifier that predicts positive on by a wide margin the majority of positive models and negative on a subset of negative models. The classifier should have a high exactness yet not a high survey. The last part is a wellspring gathering of various classifiers, with all classifiers predicting positive for the model to anticipate positive (Li, Fu, Sun, & Sun, 2018). While looking at an image, the fountain grouping tests an enormous number of sub-windows at different positions and sizes (Li, Fu, Sun, & Sun, 2018). Assuming all classifiers foresee that one sub-window is confirmed, the model denotes a thing at that sub-window's situation and size. One advantage of the overflow is that expecting all classifiers have extremely high precision and basically sufficient survey, the last classifier should have both high exactness and high audit (on a crucial level). Another colossal advantage of the wellspring is that there is convincing explanation need to execute remaining classifiers accepting one classifier predicts negative, achieving the system rapidly clearing out plain establishment sub-windows and focusing in dealing with effort on promising sub-windows. One impediment of involving Viola-Jones for seeing boats in satellite photos, as is featured in Máttyus' review, is that Viola-Jones is pivot variation (KLUYVER, et al., 2016). Máttyus conquered this issue via preparing solely on ships confronting one way and afterward rehashing the ID technique with turned input images from 16 elective points KLUYVER, et al., 2016). This extensively expands the execution time. Regardless of the absence of figures, Máttyus claims that the methodology can perceive ships with high precision in the event that the photos don't contain huge waves KLUYVER, et al., 2016).

All models for object distinguishing proof that have two phases comprise of a district idea stage and a discovery stage. The area idea step inspects the information image and predicts locales that ought to approach a thing. The subsequent step utilizes the locales with connected image information to figure likely things around the expected area.

1.2.2. R-CNN

R-CNN: Areas with CNN qualities, introduced by (Carlet & Abayowa, 2017), is the first successful (by contemporary principles) object ID approach, affecting the greater part of present day object location calculations (Liu, Liu, Huo, & Fang, 2022).The item identification approach, as the name infers, depends on region ideas and convolutional neural network. R-CNN's overall engineering is found in Figure 1.

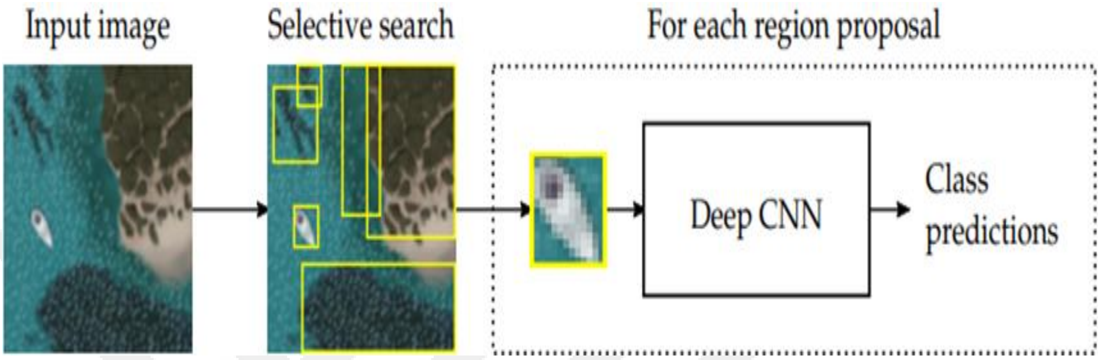


Figure 1: Overview of the R-CNN architecture (Carlet & Abayowa, 2017).

The district proposition is the model's underlying part, and it is liable for extricating regions in the information image that might contain an article (Carlet & Abayowa, 2017). In this present circumstance, a district is similar to a jumping confine that it is a square shape with a situation in the image that ought to approach a thing. They utilized particular pursuit to make around 2000 district thoughts for each image in the main review, yet they likewise included instances of a few extra techniques that may possibly propose regions (Carlet & Abayowa, 2017). Particular pursuit works by distinguishing minuscule sections in a image that are connected here and there, like tone or surface, and afterward gathering comparative portions in a base up method while additionally creating bouncing boxes around the fragments (Li, Wang, Jiang, & Chan, 2022).

The second piece of R-CNN is a convolutional cerebrum network that examinations the locale suggestion's piece of the image (Carlet & Abayowa, 2017). In R-CNN, a CNN made from 5 convolutional layers, 2 completely associated, and one outcome layer made by (FELZENSZWALB & HUTTENLOCHER, 2003). Is used, yet the last layer is subbed with an outcome layer of a size legitimate for the amount of classes (number of various groupings of things) expected (Carlet & Abayowa, 2017). The CNN was first made for object order, and it included max pooling between parts of the convolutional layers, as well as passing the consequence of each layer through a ReLU commencement capacity (FELZENSZWALB & HUTTENLOCHER, 2003). Since the CNN requires a fixed-size input image and the suggested districts contrast in size and perspective extent, image parts from the region proposals are wound to a 227x227 pixel image and dealt with into the CNN. The last part of R-CNN processes the CNN discoveries. Sifting through undesired thoughts is one model. For instance, in the event that two region recommendations have a huge cross-over and the estimates for the two districts are a similar class, just the locale with the best extended likelihood for that class is save (Carlet & Abayowa, 2017).

1.2.3. Fast R-CNN

Quick R-CNN has a comparable design to SPP-net, yet it figures out how to cut preparing time and memory use while further developing article acknowledgment precision by utilizing aperform various tasks deficit capability for preparing the entire model (barring the locale proposition) (Bradski & Kaehler, 2008). SPP-net trains every part of the model freely, which has issues, for example, not having the option to calibrate the CNN part for the article location task it is utilized for (Bradski & Kaehler, 2008). To start, the model utilizes two completely connected organizations to make forecasts. A foundation class is one of the classes that the model can foresee and ought to foresee when the district of interest comes up short on standard class (Bradski & Kaehler, 2008).

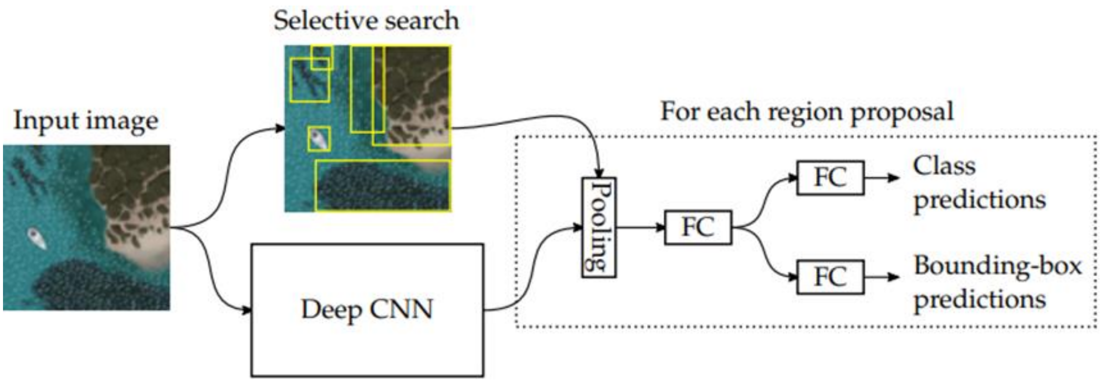


Figure 2: Architecture of Fast R-CNN (Bradski & Kaehler, 2008).

The loss function and SGD are used to prepare all parts of the model (barring the locale proposition) (Bradski & Kaehler, 2008). Smaller than usual bunches of N images are utilized, with a sum of R tested district ideas, yielding R/N locales per image. To increment preparing effectiveness, a low N = 2 and a high R = 128 are used to permit calculation and memory to be divided between regions from a similar image (Bradski & Kaehler, 2008). Figure 2 portrays the entire Quick R-CNN design. The pooling layer is like SPP-net, then again, actually rather than an entire pyramid, just a single layer with a greater matrix size (for example 7x7) is utilized (Bradski & Kaehler, 2008).

1.2.4. Faster R-CNN

Ren et al. upgrade Quick R-CNN by utilizing a district proposition organization (RPN) for locale recommendations and sharing convolutional layers used for object discovery (Li, Gu, Huang, & Wen, 2019). The RPN is very like the Quick R-CNN object identification organization. Both start with a convolutional network that creates highlight maps from which values in a particular region are recovered and taken care of into a completely associated network. The result of the completely associated network is shipped off two sister organizations, one of which predicts class probabilities and the other jumping boxes. The RPN contrasts in that the class probabilities are just "object or not thing," and anchor boxes and a sliding window are utilized as beginning core interests. The thing acknowledgment uses region suggestion as a beginning stage, however since the RPN is the part that delivers the locale proposals, it can't include them as an early phase, requiring the utilization of anchor boxes and sliding windows.

Quicker R-CNN involves square shapes with shifting sizes and angle proportions as anchor boxes. Ren et al. used three obvious scales and three unmistakable perspective extents, yielding k = 9 anchor boxes (Li, Gu, Huang, & Wen, 2019). The sliding window goes through the whole component map, eliminating three 3x3 features at the same time (per incorporate aide) and sending them to the totally associated layer (Li, Gu, Huang, & Wen, 2019). The class layer predicts 2k characteristics tending to thing or not such a great amount for each k, while the bouncing box layer predicts 4k characteristics tending to x, y, level, and width similar with the k anchor boxes (Li, Gu, Huang, & Wen, 2019). Since the RPN and object discovery share convolutional layers, the peripheral opportunity to compute area suggestions is very short (Li, Gu, Huang, & Wen, 2019). Since the district proposition is likewise a neural network, all parts in Quicker R-CNN can be prepared utilizing a similar technique (for example SGD), yet the preparation method depicted (Li, Gu, Huang, & Wen, 2019). At the point when tried on PASCAL VOC, Quicker R-CNN gets a somewhat higher Guide than Quick R-CNN, with 59.9% Guide for Quicker R-CNN contrasted with 95.03%.mAP for Quick R-CNN. (Utilizing specific hunt) (Li, Gu, Huang, & Wen, 2019). In any case, the RPN experiences issues gathering objects with outrageous structures or scales, bringing about lower exactness (Liu, Liu, Huo, & Fang, 2022). Eggert et al. support the lower execution on little things brought about by the RPN not precisely tracking down the items (Gallego, Pertusa, & Gil, 2018).

1.2.5. YOLO-v5

Redmon et al. fostered the You Just Look Once (Consequences be damned) object recognition model by considering object identification as a solitary relapse issue (Pandey, Niwaria, & Chourasia, 2020). The information image is sent straightforwardly into the organization, which comprises of 24 convolutional and two completely associated layers and predicts bouncing boxes, certainty scores, and arrangements (Pandey, Niwaria, & Chourasia, 2020). Notwithstanding, the bouncing boxes and certainty evaluations are not quite the same as the class expectations and are then converged to give a last forecast. The data image is additionally separated in an SxS network where B hopping boxes joined with conviction scores and probabilities for each class are expected for each structure cell (Kumar, Zhang, Su, & Wei, 2022). The Just go for it assumption procedure with a grid size of S = 3 and predicting a B = 1 hopping box with conviction assessments per structure cell. It ought to likewise be noticed that the information image is contracted to a set size prior to being passed to the first convolutional layer. Redmon et al. utilize a 448x488 image, a framework size of S = 7, anticipate B = 2 jumping boxes utilizing certainty scores, and anticipate 20 classes for every cell on PASCAL VOC, bringing about a 7x7x30 result (Pandey, Niwaria, & Chourasia, 2020).

1.2.6. SSD

Liu et al's. Single Shot Finder (SSD) use plan ideas from Consequences be damned and speedier R-CNN to convey a one stage object pointer that is both quicker and more careful than only let it all out and Speedier R-CNN. (Ian, Yoshua, & Aaron, 2016). On PASCAL VOC 2007, SSD with 300x300 data photos achieves 74.3% Aide while taking care of 59 images each second (Ian, Yoshua, & Aaron, 2016), however Just go for it accomplishes 63.4% Guide while handling 45 images each second (Kumar, Zhang, Su, & Wei, 2022). SSD with 512x512 info images accomplishes 87.9% Guide while handling 22 images each second, contrasted with 73.2% Guide while handling 7 images each second with Quicker R-CNN (Nie, Zhang, Niu, Dou, & Xia, 2017). The table 1. Below illustrate the summary of ship and object detection related works

Table 1: State of art comparison table

Article	Method	Performance
A Vehicle Detection Method for Aerial Image Based on YOLO (Pandey, Niwaria, & Chourasia, 2020)	This exploration incorporates coordinating three openly available datasets into a solitary dataset appropriate for preparing YOLOv3 calculations, as well as changing organization settings to work on the calculation's exhibition.	The model's exactness and not entirely set in stone to be 76.6% and 92.1%, separately..
Vehicle and Vessel Detection on Satellite Imagery: A Comparative Study on Single-Shot Detectors (Pritt & Chern, 2017).	The precision, review, and AP rules are utilized to analyze the exhibition of the YOLOv2, YOLOv3, D-Just go for it, and YOLT calculations. The calculations have been upgraded to guarantee ideal precision.	As per the creators, D-Just go for it has a most extreme exactness of 60% for cars and 66% for ships (vessels).
Ship Detection in Optical Satellite Images via Directional	The scientists made a two-organized CNN-based transport distinguishing proof calculation	The proposed method's detection accuracy for small

Bounding Boxes Based on Ship Center and Orientation Prediction (Maity, Banerjee, & Chaudhuri, 2021).	that spotlights on the delivery community and image direction. They contrasted their strategy with state of the art location calculations like Quicker R-CNN, SSD, and Just go for it and verified that the new technique outperforms the others.	ships can be satisfactory, with mAP of around 81%.
New Approaches and Tools for Ship Detection in Optical Satellite Imagery (Cordova, Quispe, Inca, Choquehuayta, & Gutierrez, 2020)	This paper presents the consequences of the YOLT and YOLOv4 profound learning calculations on two datasets.	On the MSDS dataset, the accuracy for YOLO and YOLOT was determined to be 69.8% and 95.94%, respectively.
Ship Detection Using Transfer Learned Single Shot Multi Box Detector (Nie, Zhang, Niu, Dou, & Xia, 2017)	For transport identification, the specialists proposed an exchange learning Single-shot Multibox Identifier (SSD).	The experiment proved that this approach has an accuracy of 87.9% and can run at 47FPS, thereby meeting the real-time requirements.
R-CNN-Based Ship Detectionfrom High Resolution Remote Sensing Imagery (Zhang, Wu, Xu, Wang, & Sun, 2019).	The creators of this work further developed the first CNN design to make a changed form of the Quicker R-CNN and afterward contrasted it with unmistakable calculations like SSD and YOLOv2.	There are 707 ships removed from a total of 744 ships in the preprocessing stage, with a 95.03% recall.

Aircraft Detection of High- Resolution Remote Sensing Image Based on Faster R- CNN Model and SSD Model (Redmon & Farhadi, 2018)	In this review, the researchers utilized Quicker R-CNN and SSD calculations to prepare on high-goal remote detecting images. As per the discoveries, speedier R-CNN is better for discovery in complex circumstances, though SSD is better for identification in single scenes. As far as in general precision, quicker R-CNN bests SSD.	Accuracy: Faster R-CNN= 98.61% SSD= 98.57%

1.3. Challenges in Ship Detection

Computer vision is an emerging field with various applications in military, social, and industrial sectors. It is progressively utilized for undertakings, for example, astute video observation, security checking, content-based image recovery, and individual re-ID. One of the critical assignments in PC vision is object discovery, which includes finding and arranging objects in images or video outlines caught through remote detecting. However, such object detection tasks pose significant challenges due to variations in the quantity, position, form, and size of objects, as well as identifying them against a varying background. Advanced algorithms and techniques are therefore required to address these challenges and achieve accurate and reliable results in object detection.

Ship detection is a specific case of object detection. In this case, remote sensing-based image quality is often degraded by daylight, mists, clouds, ocean waves, and other backgrounds, such as islands, harbors, and coastlines. The quality of the images is affected by three main factors: weather conditions, imaging conditions, and target properties. Weather conditions include sea state and cloud conditions. For example, sea states such as calm or stormy, with large waves, can affect the imaging. Cloud conditions vary from no clouds to thin or thick clouds, which can affect the grey scale distribution of the image and the target-ship features. Imaging conditions, such as lighting and weather seasons, can also affect the brightness and contrast of the image, impacting the image quality.

The target property of ship detection algorithms includes the detection of ships moving across a body of water, which can have variable wakes based on the size, speed, sea states and weather conditions. The algorithms must be able to handle scale and rotation changes, as ships vary in size and can move in rotational patterns. Low contrast can also create difficulties in properly detecting ships, and false ship detections may occur due to broken clouds, sea waves, ship wakes, islands and other background objects.

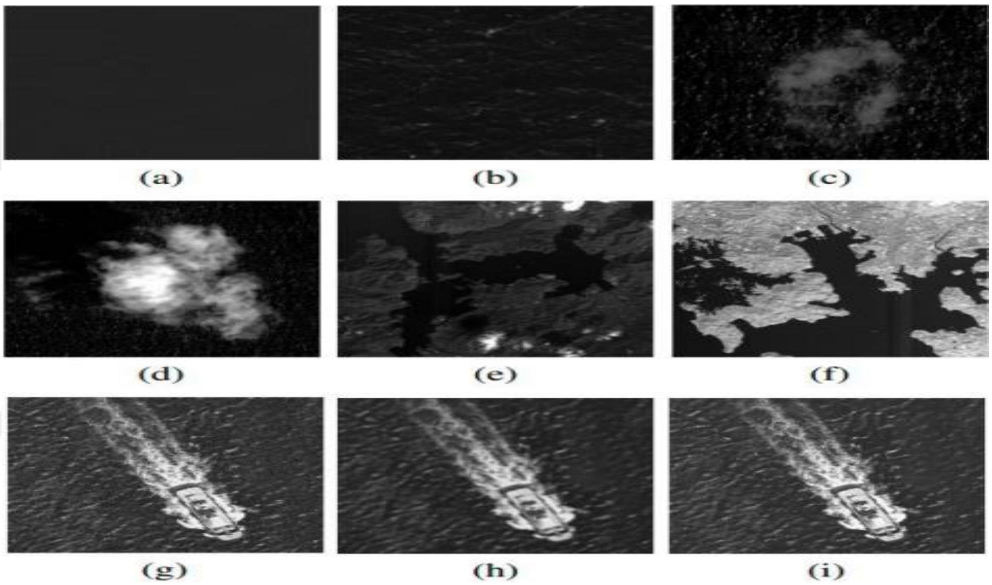


Figure 3: Optical remote sensing images under different conditions.

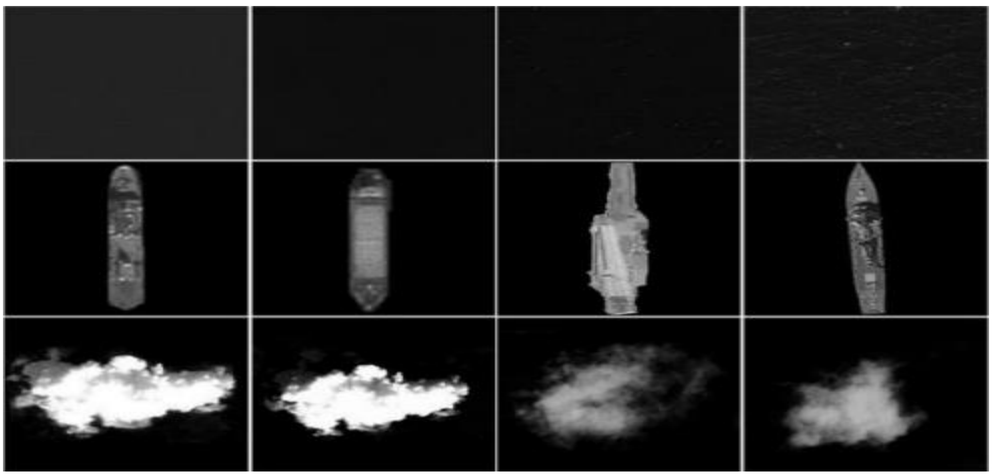


Figure 4: Sea surface, ship target and cloud samples

The transportation of harmful algae via ships can have a significant negative impact on coastal waters in Western British Columbia, due to the introduction of new organisms to the ecosystem. Algae blooms can occur when the introduced algae exceed the carrying capacity of the environment, leading to the depletion of oxygen and other resources. This can result in the death of marine organisms and disrupt the fragile balance of the local ecosystem. Additionally, the introduction of new species can also lead to the spread of disease or the displacement of local species, resulting in an overall decrease in biodiversity. In recent years, harmful algal blooms (HABs) have become increasingly frequent in coastal areas, putting both ecological and socio-economic systems at risk. Approximately 100 microalgae species are known to produce toxins, with Domoic Acid (DA) and Paralytic Shellfish Poisoning (PSP) being the most commonly found during the spring and summer seasons These toxins can be hazardous to humans and animals (Feng, Zhao, & Kita), (Fujino, Hatanaka, Mori, & Matsumoto, 2019). In 1793, a case of paralytic shellfish poisoning (PSP) was documented at Poison Cove, resulting in five people becoming ill and one death. Symptoms included neurological issues such as paralysis or respiratory arrest. This was the first recorded instance of PSP, which was linked to contaminated mussels from the area. Since then, other species of shellfish have been found to contain the toxin, domoic acid (DA), which has become increasingly widespread (FELZENSZWALB & HUTTENLOCHER, 2003). Even though no illnesses have been reported since 2005 due to the effective biotoxin monitoring program, which has closed down oceanic agriculture and fisheries, the most prevalent paralytic shellfish toxins (PST) in B.C. are still those of Alexandrium blooms (FELZENSZWALB & HUTTENLOCHER, 2003). This has caused potential bankruptcy, layoffs, and lost harvests (FELZENSZWALB & HUTTENLOCHER, 2003). Additionally, toxins can remain in seafood, even after toxin levels have decreased in the surrounding water (Evans, Al-Sahaf, Xue, & Zhang, 2018). The buildup of toxins in local sea life, such as fish, seals, whales, etc., is of particular concern due to the fact that these biotoxins cannot be destroyed through cooking or processing of seafood, meaning that they can enter the human food chain (Evans, Al-Sahaf, Xue, & Zhang, 2018).

Satellite data is collected through a swath process, which allows for the coverage of large areas of ground in a single image as the satellite passes overhead (Alibrahim & Ludwig, 2021). There are various types of satellites, including multi-spectral, hyperspectral, and panchromatic (Alibrahim & Ludwig, 2021). These sensors can range from basic registrations of the red, green, and blue (RGB) channels, as typically seen in optical imagery, to complex multi-spectral sensors that span multiple bandwidths of the light spectrum.

A system is considered centralized when it contains a single primary node or process that delegates tasks to other nodes, referred to as workers. Decentralized distributed systems offer a more complex view of primary/worker relations. A node may be a primary of a centralized subset of the network and simultaneously exist as a worker of the larger network, or it may exist as a worker of the larger network while also having its own pool of workers. In a cloud based environment, a hierarchical distributed system architecture is typically used. This architecture offers a tiered layer, where a node can communicate with its predecessor or successor, either one level above or below. This type of architecture should be taken into account when implementing a scheduler. Not only should the architecture's connectivity of primary, workers, and resources be considered, but also the policies and access controls associated with the environment. These access controls limit the network and physical resources that a worker can communicate with and should be taken into consideration when creating the scheduler. Furthermore, the organization of the architecture may play a role in the scope of the scheduler, as it can affect the communication between nodes in order to achieve the system goal.

1.4. Problem Statemant

The detection of ships in satellite images is an essential task in maritime surveillance and security. Various deep learning techniques have been employed to achieve accurate and efficient detection, including Artificial Neural Networks (ANNs), Convolutional Neural Networks (CNNs), and You Only Look Once (YOLO) object detection framework. However, the performance of CNNs is heavily reliant on the architecture of the base network, and hyperparameter optimization remains a critical challenge, often addressed through manual tweaking or other optimization methods. While recent literature has introduced the use of evolutionary algorithms to optimize hyperparameters for object detection in satellite images, this area remains underexplored. Therefore, this study aims to compare and evaluate the performance of ANNs, CNNs, and YOLO using hyperparameter optimization with evolutionary algorithms for ship detection in satellite images, addressing the research gap in this field.

1.5. Aims and Objctives

The main aim of this research is to compare and evaluate the performance of Artificial Neural Networks (ANNs), Convolutional Neural Networks (CNNs), and You Only Look Once (YOLO) object detection framework for ship detection in satellite images. Specifically, the objectives of this study are:

i- To investigate the effectiveness of ANNs, CNNs, and YOLO for detecting ships in satellite images.

ii- To optimize hyperparameters using evolutionary algorithms for ANNs, CNNs, and YOLO to improve detection accuracy.

iii- To improve the accuracy and reliability of ship detection in such images. ANNs and CNNs are used for classification, segmentation, and detection in remote sensing images, including image data from satellites used for ship detection.

iv- To develop advanced ship detection models that are robust to variations in weather conditions, lighting, and other factors, and can provide accurate and reliable detection of ships in regions of interest

The overall goal of this research is to provide insights into the strengths and weaknesses of different machine learning approaches for ship detection in satellite images and to propose an optimized and effective method for accurate and efficient detection.

1.6. Contribution

The contribution of this master research is two-fold. Firstly, this study proposes an optimized deep learning approach for ship detection in high-resolution satellite and aerial data. By tuning the hyperparameters of ANNs, CNNs, and YOLO using a genetic algorithm, the newly created models are expected to outperform the original deep learning model in terms of accuracy, precision, recall, and F1-score. The use of evolutionary algorithms to optimize hyperparameters is a novel approach in this field and can significantly improve the efficiency and effectiveness of ship detection in satellite images.

Secondly, this research aims to reduce the manual process of identifying objects from satellite imagery, which is a time-consuming and labor-intensive task. The proposed deep learning models can accurately detect and localize ships in satellite images, which can significantly reduce the workload of experts in the field. This can benefit various applications, including meteorology, map making, environmental monitoring, policing, and disaster response, by providing timely and accurate information for decision-making.

Overall, this research can contribute to the development of efficient and effective approaches for ship detection in satellite and aerial images, which can have significant implications for various fields and applications.

1.7. Outline

• Section 1, which is a prologue to the issue and portrays the issue explanation, points and targets, and examination questions that are proposed to arrive at the review's objective.

• Section 2 gives a total scenery of all the information expected to comprehend and direct the examination study.

• The accompanying section, Strategy, gives a full rundown of the whole examination plan. The methodology is isolated into two sections: a survey of the writing and an examination to respond to the exploration questions.

• The consequences of the writing survey and the examinations acted in the review are all given in Section 5.

CHAPTER TWO

LITERATURE REVIEW

Gives a total scenery of all the information expected to comprehend and direct the examination study.

2.1. Machine Learning

The logical investigation of calculations and factual models that a framework utilizes to do a specific errand without being expressly modified is alluded to as AI (MAO, SUN, LIU, & JIA, 2019). It is gotten from counterfeit cognizance (man-made brainpower), and it gives computers the capacity to think and learn without the help of any outer sources. AI can on the other hand be characterized as how a framework adjusts its activities to upgrade the general exactness of a specific undertaking, where precision is characterized as the times the framework chooses precise arrangements (Alzubi1,, Nayyar, & Kumar, 2018). AI approaches are regularly arranged into four gatherings in view of calculation choice. There are four kinds of learning: managed learning, solo learning, semi-administered learning, and support learning. (Saravanan & Sujatha, 2019).

2.1.1. Supervised Learning

It is a kind of advancing in which the calculation gets data from named earlier and current information. The educational experience starts with the dataset's preparation, and from the preparation, the framework can develop a capability to foresee yield values. With enough practice, the framework will actually want to recognize blames and change its ways of behaving to more readily foresee yields. At last, the model works on itself relying upon the outcomes (Saravanan & Sujatha, 2019).

2.1.2. Unsupervised Learning

These AI strategies join administered and solo learning techniques. This method is exceptionally successful when the information is unlabeled and the most common way of naming the information is tedious or needs proficient human experience. Semi-directed learning is utilized to tackle grouping, relapse, and forecast issues (Alzubi1,, Nayyar, & Kumar, 2018).

2.1.3. Semi-Supervised Learning

These AI strategies join administered and solo learning techniques. This method is exceptionally successful when the information is unlabeled and the most common way of naming the information is tedious or needs proficient human experience. Semi-directed learning is utilized to tackle grouping, relapse, and forecast issues (Alzubi1,, Nayyar, & Kumar, 2018).

2.1.4. Reinforcement Learning

Support learning is a subfield of AI in which a specialist makes moves in a climate and gets criticism from the climate. Criticism could appear as remunerations or issues. The specialist endeavors to do exercises that amplify the award, and ideal way of behaving is recorded to further develop execution (MAO, SUN, LIU, & JIA, 2019).

2.2. Neural Networks Basics

Both completely associated (FC) neural network and convolutional neural network (CNN) are Artificial Neural Network (ANN) intended to take some info and produce an ideal result (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014). The organizations are normally worked by a grouping of increases, augmentations and applying nonlinear capabilities (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014). Figure 2.1 outlines a minuscule FC network where every hub in a layer is subject to all qualities in the past layer, for example y1 in the subsequent layer is subject to all x qualities. The loads and inclinations (for example w separately b in the figure) is the qualities changed to make the organization produce an ideal result. The cycle is depicted in Segment 2.2. The capability an is a nonlinear capability that will be examined further in this segment. CNN is one sort of ANN that is proficient and functions admirably with image input information (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014). Figure 6 portrays a little CNN with only one convolutional layer and one max pooling layer. The convolutional layer capabilities similarly as discrete convolution does. A part (w in the image) is sliding over the info (x in the figure), processing the spot result of the piece and the piece of the information window over which the bit is sliding. The result is likewise determined utilizing a nonlinear capability and predisposition (an and b in the delineation). The organization in the figure likewise incorporates a maximum pooling layer with a size and step of 2, demonstrating that two qualities are handled simultaneously and that the handling moves two information values for each result esteem. Pooling is a proficient technique for bringing down the quantity of values in a CNN.

The convolutional layer and the FC layer fluctuate in that the convolutional layer normally contains a lot less loads than the FC layer for similar number of information values (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014). While processing unmistakable result values for fluctuated input esteems, the piece values stay steady (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014).

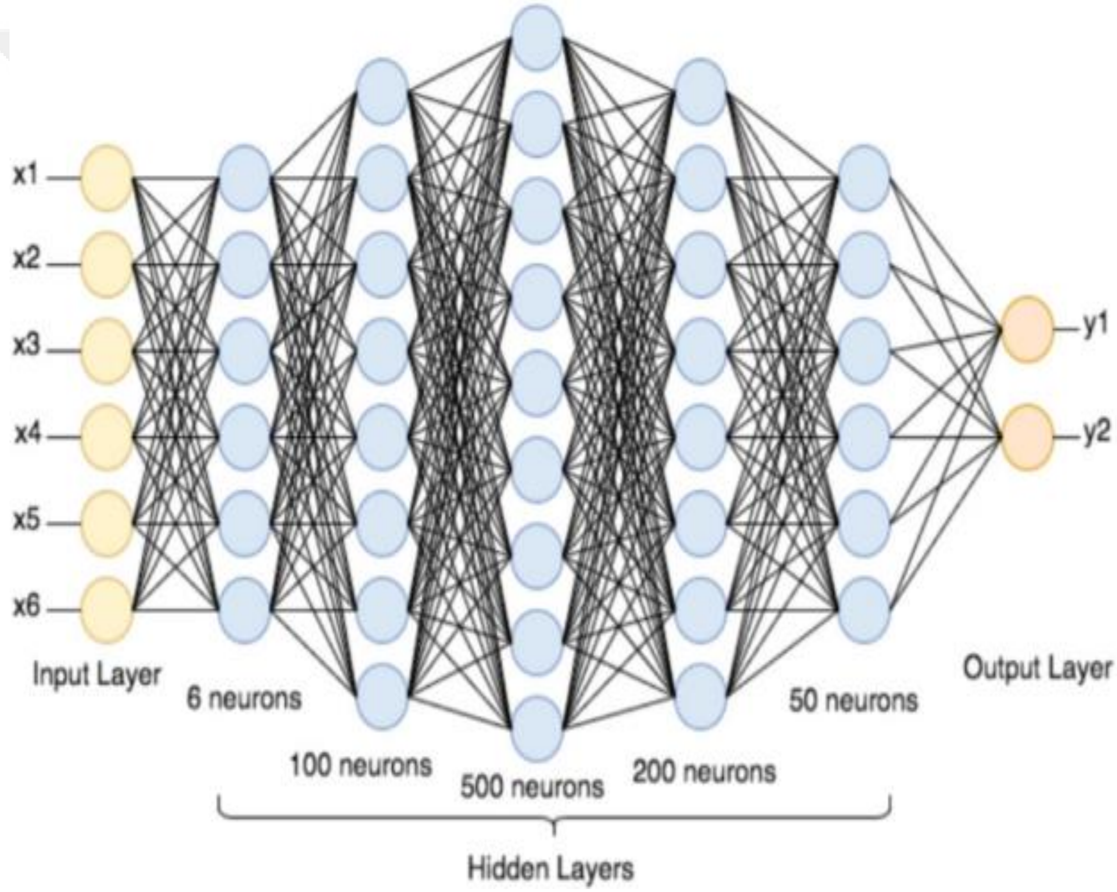


Figure 5: Deep Neural Network Architecture (Bahi & Batouche, 2018).

The deep neural network configuration is portrayed in Figure 5, which is a progression of completely associated layers with various secret players between the information and result layers.

It is normal to utilize two-layered CNN while working with two-layered images. Two-layered convolutional layers capability comparably to one-layered convolutional layers, yet in two aspects. The part is two-layered and moves in two aspects to cover the entire information (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014). Moreover, every pixel in a RGB image has three qualities for red, green, and blue. A RGB image can be seen as three-layered contribution somewhat, albeit one could contend that it's anything but a third aspect in light of the fact that the general request of red, green, and blue doesn't make any difference and that they are rather various numbers addressing a similar pixel. Figure 2.3 shows a convolutional layer applied to a RGB image with just two components of convolution while catching information from all tints of the image. It is significant that the portions have autonomous qualities for each tint, permitting the organization to independently treat different varieties. Figure 6 likewise tells the best way to utilize a few pieces in a convolutional layer. More pieces bring about additional information yield. By utilizing various portions, every piece might catch particular components of the approaching information (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014). A convolutional layer's result has a similar design as the info image. The goal of the info image influences the initial two aspects, and the place valuable is connected to a point in the information image. Moving something in the info image along the two aspects will set off a similar development in the result. The "third aspect" has values depicting a similar district, with no geological data held by the succession of the qualities (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014). On the off chance that the Figure 6 result were used as contribution to another convolutional layer, a part size of, say, 2x2x4 may be utilized to permit every bit to involve information from all pieces in the former layer.

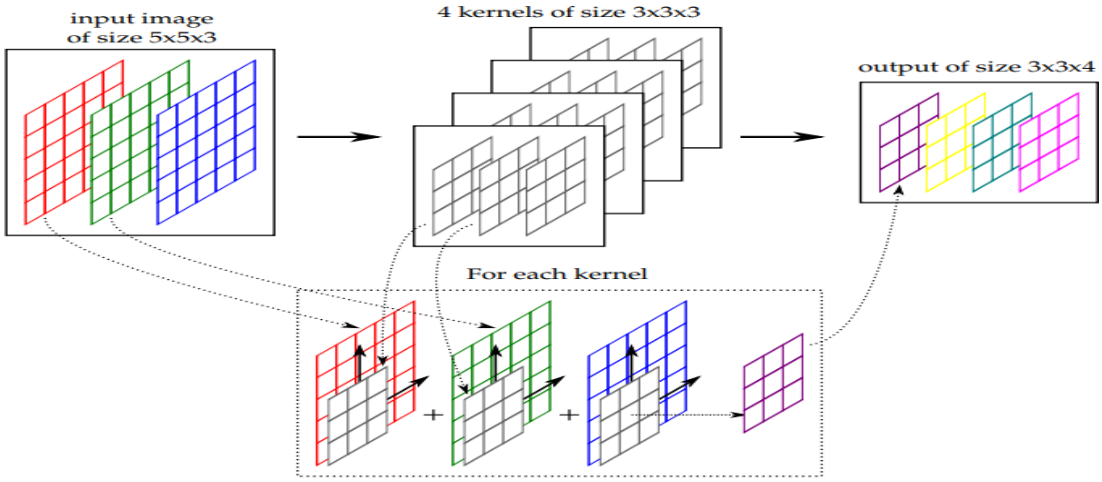


Figure 6: Illustration of a convolutional layer for two-dimensional RGB images (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014).

Changing the organization's loads and predispositions to make wanted yields for related input preparing neural network involves. Images give contribution to protest identification, and the ground truth bouncing boxes and names for each image lay out the expected result for that image. Since the organizations are so enormous, there are such countless factors to change that testing every one of them is incomprehensible. There is no known way for straightforwardly registering an ideal. The preparation methodology examined in this segment starts for certain underlying upsides of the factors to tune (which can be randomized) and afterward iteratively changes the factors to upgrade the organization (Chen, Xiang, -Lin Liu,, & -Hong Pan, 2014).

2.3. Related work

This section contains comparable exploration in the subject of identifying planes, vehicles, and boats from satellite and flying photography, as well as offering significant bits of knowledge into the hyperparameter tuning of the Deep learning approaches tended to by the proposition.

While different item recognition calculations have previously been canvassed in this part, this segment rapidly addresses a couple of others. The principal justification for not exploring any of these techniques is to restrict the extent of this review. Veil R-CNN is a model that expands Quicker R-CNN by delivering a pixel cover for every expectation (Claesen & Moor, 2015). A pixel veil tells which pixel in the image the thing covers, while jumping boxes determine a square shape that denotes the item. While a veil can be helpful at times, it very well might be inadequate for recognizing a boat in a satellite image. While simply considering jumping boxes AP, Veil R-CNN beats Quicker R-CNN (Claesen & Moor, 2015). Another two-stage object indicator is the district based Completely Convolutional organization (R-FCN), which is totally convolutional (Alzubi1,, Nayyar, & Kumar, 2018). The critical advantage of R-FCN over speedier R-CNN is the faster examination time while conveying identification aftereffects of similar quality (Alzubi1,, Nayyar, & Kumar, 2018).

Redmon and Farhadi made a second and third better rendition of Just go for it (Laban, Abdellatif, Ebeid,, Shedeed, & Tolba, 2020) (Lee, Parka, & Sima, 2018). YOLOv2 highlights different design upgrades that make it speedier and more precise than Consequences be damned (Laban, Abdellatif, Ebeid,, Shedeed, & Tolba, 2020). They likewise permit YOLOv2 (Laban, Abdellatif, Ebeid,, Shedeed, & Tolba, 2020). To conjecture north of 9000 unmistakable classes, which may not be as advantageous while simply spotting ships in satellite photographs. YOLOv3 just incorporates a couple of upgrades, however it actually has a huge lift in location quality for little items (Lee, Parka, & Sima, 2018). This builds YOLOv3's capacity for spotting ships in satellite photographs when contrasted with YOLOv2 and Consequences be damned. RefineDet by Zhang et al. is another stage object locator that has two modules like two phase object finders that improve anchor boxes and distinguish objects (Lin, Shi, & Zou, 2017). RefineDet beats SSD (Lin, Shi, & Zou, 2017) in distinguishing little articles, making it a model worth examining for use in identifying ships on satellite photographs. While working with turn invariant satellite photographs, pivoted bouncing boxes (RBB) may fit better compared to standard jumping boxes that are rarely pivoted. Liu et al. propose a rotational district based CNN that predicts RBB in satellite images for transport recognizable proof (Guo, et al., 2020). The model comprises of two phases: a rotational district of interest pooling stage and a RBB with class indicator (Guo, et al., 2020). Ding et al. propose an area of interest transformer that turns the prescribed locales to match the recognized things better (Ammar, Koubaa, Ahmed, Saad, & Benjdira). It ought to be noticed that these models require a dataset including RBB for preparing and evaluation. In the examinations directed by the creators, both the pivoting locale base CNN and the R-CNN with the area of interest change performed well (Ammar, Koubaa, Ahmed, Saad, & Benjdira) (Guo, et al., 2020).

CHAPTER THREE

METHODOLOGY

The accompanying section, Strategy, gives a full rundown of the whole examination plan. The methodology is isolated into two sections: a survey of the writing and an examination to respond to the exploration questions.

3.1. Introduction

The methods proposed for this thesis work, as well as the overall conceptual mechanism, are described in the first section of this chapter. The subsequent segment adjusts chosen designs and proposes new models, as well as calibrating and hyperparameter setting. The third segment is worried about the strategy's execution viewpoints, and it depicts calculation plans for imagining the result of the models planned, as well as customization to prepare on the proposal dataset.

To assemble bits of knowledge and data from all earlier examinations in the space of accentuation, a writing survey is essential. Articles and examination papers are considered as a component of our momentum research. The objective of the writing study is to pick a bunch of Deep learning draws near. Different decisions incorporate an orderly writing survey (SLR), but in light of the fact that it requires an excessive number of assets, the cycle was diminished to a writing survey.

Besides, compounding strategies were used to distinguish more important papers that could have slipped through the cracks in any case. In reverse Compounding and Forward Compounding are the compounding strategies utilized. From the get go, few articles are picked, and they are consequently presented to compounding.

3.2. Overview

In our current days, the sky is filled with drones, and the space with satellites. Those drones and satellites are equipped with high resolution digital cameras, capable of capturing images and videos of high definition. These images and videos are very useful for a variety of application such us traffic monitoring, wild fires early detection, urban city planning, high precision agriculture and many security oriented services. However, their large abundance makes it almost impossible for humans to manually inspect them and take decisions based on them. Therefore, nowadays the industry, working together with the scientific community, has employed artificial intelligence and computer vision techniques in order to automate the process of extracting knowledge from such images. In this work, the focus is on the detection of ships captured from satellite images. Developing algorithms able of detecting ships can results to numerous important applications that can improve trade, security and defence. Such applications are ship monitoring, piracy confrontation, accidents prevention, surveillance, spying, etc.

A lot of work has been done in the filed of urban planning, where they employ semantic segmentation models to extract building footprints from satellite images. Such examples are (Claesen & Moor, 2015). And (Ammar, Koubaa, Ahmed, Saad, & Benjdira), where in both cases they have used some variations of the U-Net model, which is a fully convolutional network. Furthermore a lot of work has been done in the field of agriculture where they apply deep learning techniques in satellite images for many purposes. For example many companies such as the company Planet5 specializes on monitoring crops growth, while also detect anomalies and try to prevent plants' destruction. Additionally, deep learning is also used in crop type classification from satellite images (Bahi & Batouche, 2018). Is a fascinating concentrate wherein they prepared Deep learning models to foresee overview based appraisals of resource abundance across 20,000 African towns utilizing openly accessible multi-unearthly satellite symbolism to decide the monetary prosperity of African nations. These neighborhood level proportions of human prosperity are basic for illuminating taxpayer supported organization conveyance and strategy choices. This approach is revolutionary as the traditional data collection methods were expensive and rare, while most African households appears on average at least weekly in cloud-free imagery from multiple satellite-based sensors.

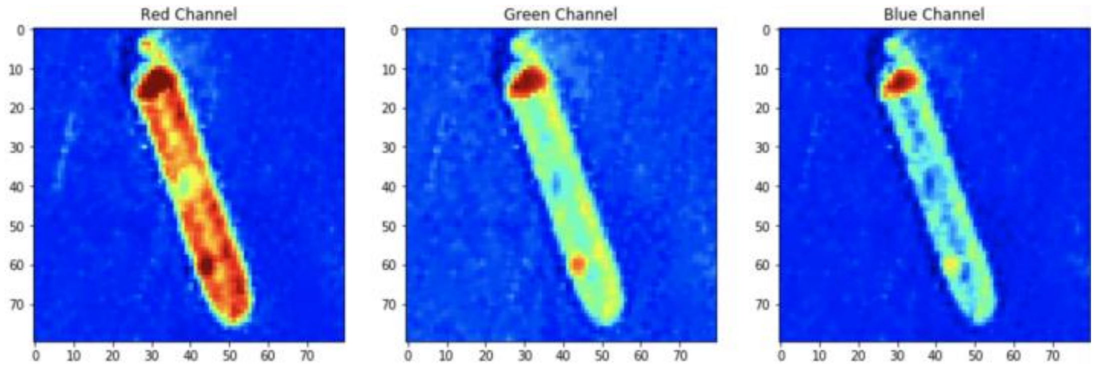


Figure 7: Image with its Mask

Extreme Earth is a Horizon 2020 project that focuses on the development of Artificial Intelligence and Big Information advancements with the capacity of increasing to the outrageous size of Copernicus information. Outrageous Earth involves these advancements in two of the European Space Organization's topical abuse stages: one for food security and one for polar districts. Its will probably make procedures and programming that will permit data and information to be separated from a lot of Copernicus information utilizing Deep learning strategies and outrageous geospatial investigation. Regarding the polar use case, the goal is to interlink the in-situ ice observations with satellite images in order to build training sets with satellite images associated with high quality ground observations, and deep learning models able to determine for the state of ice in the polar cycle. Ice monitoring is a very crucial task for the safely navigation through the Arctic, especially nowadays where the ship traffic in the Arctic has increased.

3.3. Proposed System

In this project the goal is to predict the segmentation masks of ships from satellite images. For this purpose, some state-of-the-art semantic segmentation models are employed, which are as follows:

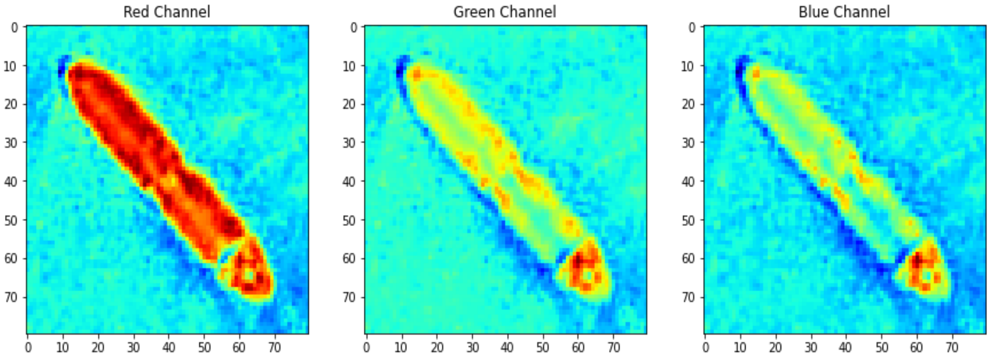
- ANN
- CNN
- YOLO-v5

The input dataset consists of satellite images capturing the sea, and a CSV Document containing the masks of each image in the form of RLE. In my approach, the images and their masks are loaded through a python generator which is forwarded to an image augmentation generator. To further pre-process the images, their values are normalized to the range [0, 1], downscaled, and the ship distribution in the images is rebalanced by under-sampling the ships-free images and oversampling the images with ships.

3.3.1. Ship Detection Dataset

The shipsnet ship image dataset is a collection of satellite images that includes both pictures with ships and pictures without ships. It is formatted as a JSON text record named shipsnet.json and is included with the dataset. The dataset is frequently used in training machine learning models to detect ships in satellite imagery. The shipsnet ship image dataset is available on Kaggle, a platform for data science and machine learning. It was originally created by R. J. Rhammell and consists of 80x80 pixel RGB image chips extracted from Planet satellite imagery.

In the dataset, there are satellite images that incorporate images with ships as well as images without ships. The two classes, notwithstanding, are similarly circulated. Kaggle is giving us our dataset. At the point when the boats were straightforwardly close to one another, there was a slight cross-over of item sections in the images containing the boats. Ships inside and across images might fluctuate in size (now and again altogether) and might be situated adrift, on moors, marinas, or somewhere else. Image chips derived from Planet satellite photos gathered over the San Francisco Bay Area comprise the dataset. It has 4000 80×80 RGB photos, 1000 of which are ship images labeled as "ship", while the 3000 left are noship images labeled as "no ship". Image chips were created using PlanetScope full-frame visual scene products that were orthorectified to a pixel size of 3 meters (Alghazo, Bashar, Latif, & Zikria, 2021).



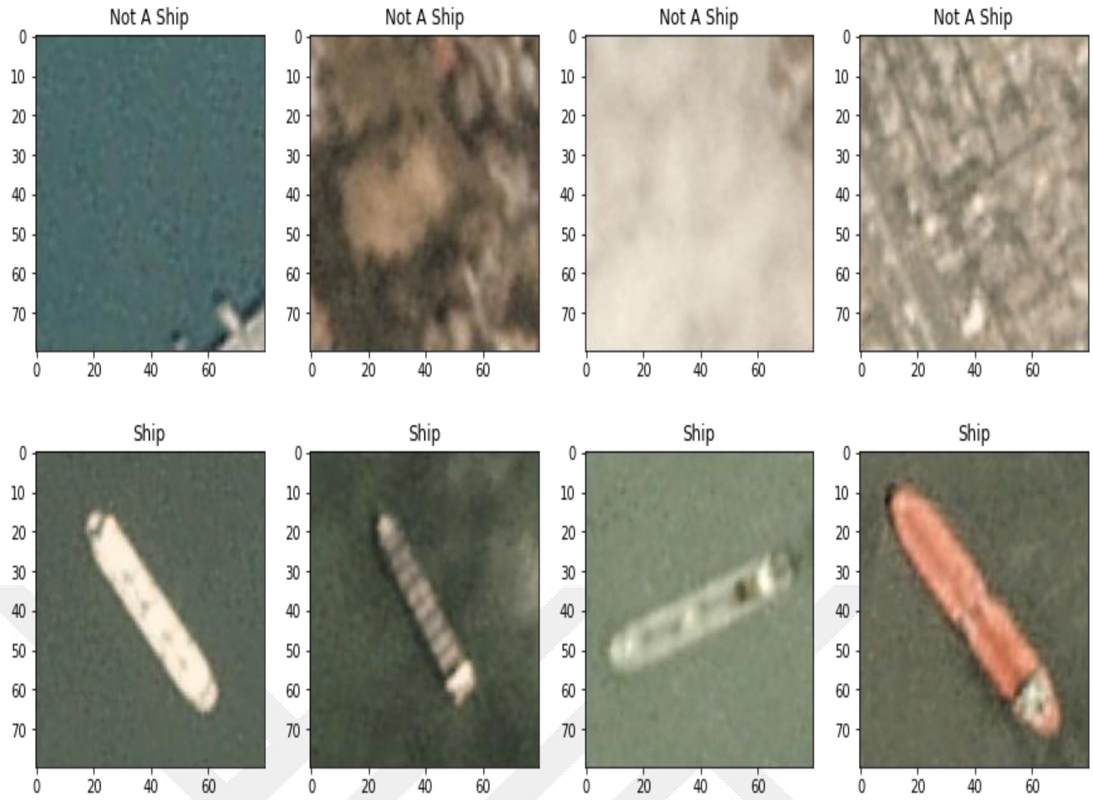


Figure 8: An example of the dataset Visualization *(*Web link to the dataset: https://www.kaggle.com/c/airbus-ship-detection/data)

3.3.2. Data Re-balancing

The very first thing noticed when downloading the data was that there was a great imbalance between the images containing the ships and those that did not. The large majority of the images consisted of plain images of the sea. Figure 3.4 depicts the number of ships in a random sample of the images, and as can be seen, the data were very skewed and needed to be rebalanced. The under-represented images containing ships were oversampled by replicating them, while the ships-free images were under-sampled by keeping just a small fraction. This way, the data were adjusted to a more preferable distribution, which is displayed in Figure 8. However, this procedure was developed before executing larger experiments consisting of more images. The images without ships were significantly more than those with ships, and due to the lack of available resources and the need to deescalate the experiments' execution time, it was decided to remove them from the training procedure. However, it is believed that this does not negatively impact the overall performance of the models, as the parts of the sea are well represented by the images that do contain ships.

This re-balancing strategy is implemented inside a data loader class, which its primary purpose is to return an image with all of its masks as one. Its first task is to read the CSV document and aggregate all the masks of the images, so each record will contain all the masks of the image it points to. Another function of this class is to split the dataset into training set and validation set, based on a given ratio. During this split the re-balancing of the images is taking place.

It is very important to mention that the data loader never loads all the images in the memory. The training and the validation set it returns, consist of records that point to the corresponding image, and the set that is provided to the models is a python generator which loads batches of images only when they are requested and right after they are no longer needed, they are removed from the memory. By applying this procedure, the memory footprint was minimized, which is essential when working with images. During the images loading procedure, their masks were decoded and merged into a single one, and the values of the image were normalized in the range of [0, 1]. Both the image and mask were down-scaled to the shape (256, 256). This down-scaling may have a negative impact on the performance of the models, but as previously mentioned, it was necessary to adapt to the available resources. Additionally, there is the option of applying filters to the images to enhance their contrast and make their edges more distinct.

This produced generator that loads images on the fly, is then forwarded not directly to the model but to another generator responsible of generating more images through the process of data augmentation. This generates multiple images from a single image by randomly applying shifting horizontally and vertically, rotating, flipping, and zooming to the initial image.

3.3.3. Data Pre-processing and Augmentation

Since the strategy means to utilize Deep learning procedures, explicitly Deep convolutional networks, which should learn elements of the provided images by removing their properties and involving them for future expectation, this study utilizes as little pre-processing as conceivable to furnish certifiable situations related with the image datasets; pre-processing methods applied to all information include: rescaling the images in the scope of 0-1 to ease image handling; Both of these means are done inside a Deep learning climate, with no work done outside of it.

3.3.4. Data analysis

Images are provided as 19200 1 vectors in this dataset. To make the image dataset a 2800x38080 4-dimension numpy array, each vector was turned into a 38080 RGB vector. • Labels: labels were converted to a 28001 1-dimensional numpy array, with 700 1's corresponding to ship photos and 2100 0's corresponding to no-ship images.

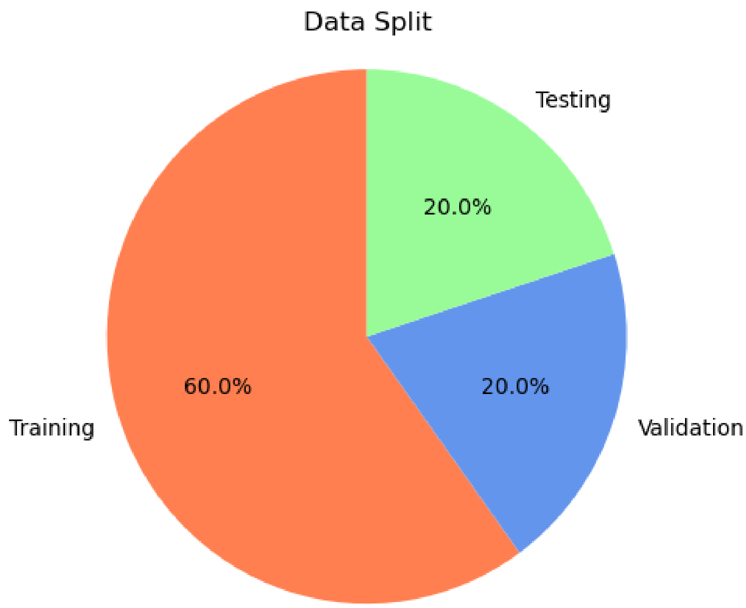


Figure 9: distribution of empty and not empty masks

According to the charts more than 75% of the images has no ships on it. If the fact that the ships take up only a very small area of the images is also factored in (as discussed in detail below), it can be concluded that the two classes of this dataset are extremely unbalanced (the ratio is ~1: 10 000). If all the images with no ships are dropped, this ratio improves slightly (to ~1: 1000), also the training time reduces significantly.

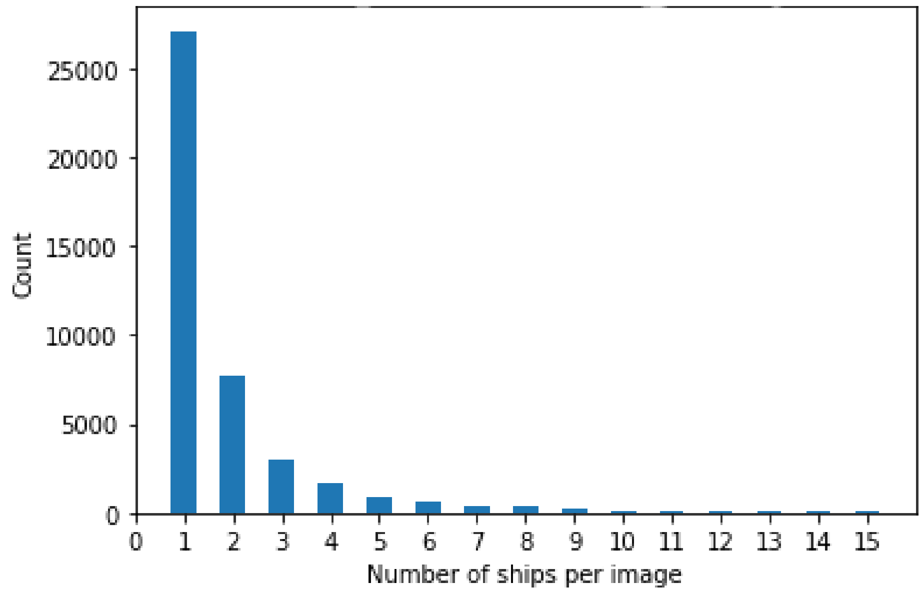


Figure 10: histogram of ship count per image

On figure 10 you can see a histogram about the number of ships on images. You can see most of the images with ships contain only a few ships. it can make the training difficult. The bar representing the number of images without ships (0 ships) is not on the diagram, because it would make even more bars invisibly small.

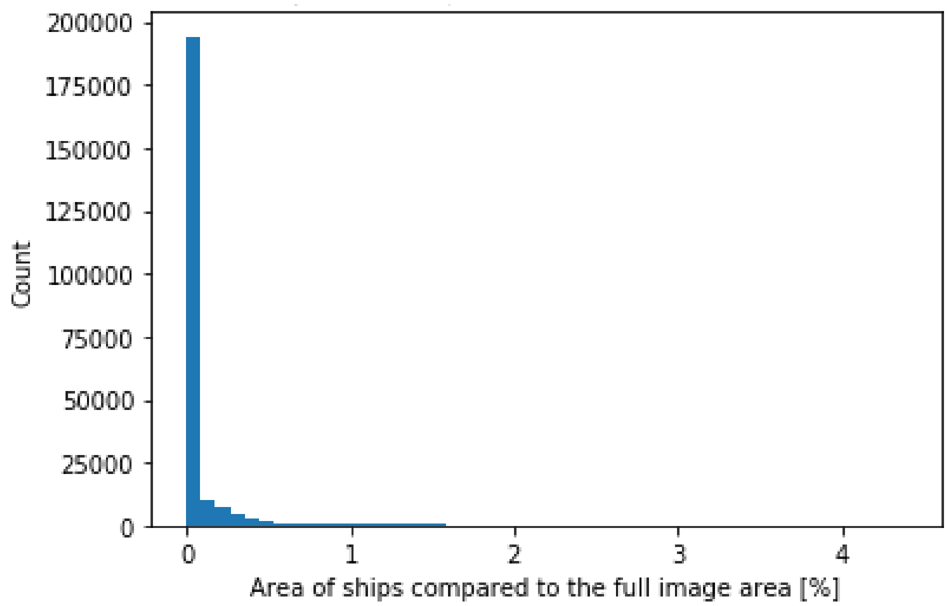


Figure 11: Ship sizes

On figure 11 you can see the ships are very small compare to the whole image (note that the vertical axis is in % of the full image area!). So this is another reason why training the network could be difficult. On the next images there are the ships and the decoded RLE masks on them.

3.3.5. Model Selection

Convolutional Neural organizations are progressive neural organization frameworks that utilization neurons like those found in the human visual cortex to make associations between the organization's different layers, which are ordinarily alluded to as convolutional and examining layers. These are feed forward Deep learning neural organization calculations prepared utilizing back-engendering strategies, and they comprise of three fundamental layers: an information layer with a characterized size, stowed away layers with numerous convolutional layers, actuation capacities, pooling layers, completely associated layers, and a last result layer (completely associated) showing the result of info layers handled and prepared with neural organizations.

Building a model using a Convolution Neural Network would have been the best choice for our problem since it falls under the domain of computer vision. The introduction of CNNs marks a pivotal moment in object detection history, as nearly all modern systems use CNNs in some form or other. The decision was made to build a ConvNet from scratch instead of selecting a pre-trained model in order to maximize learning from the project while keeping the model simple to understand. To understand this model, an example is considered where an RGB image with the dimension 1000 X 1000 needs to be processed.

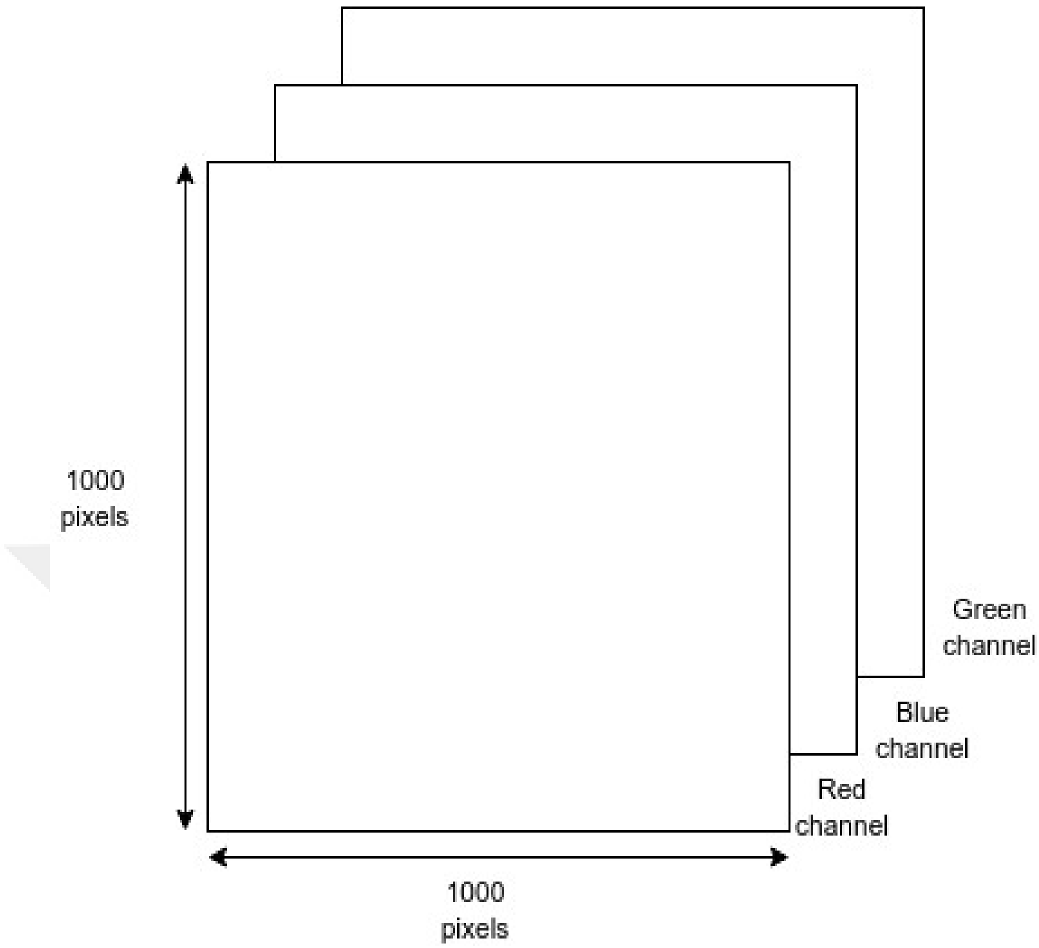


Figure 12: RGB image with the dimension 1000 X 1000

When a deep neural network is created, the input vector to the neural network will have 3 million input values (1000x1000x3), and if there are only 1000 nodes in the first hidden layer , we have not more, the numbers of parameters to learn in the first transition itself will be around 3 million * 1000 = 3 Billion and optimizing these much numbers of parameters is not feasible at all and this is just for a small image in the only first transition, hence we need to have a practical model to deal with this problem. This is where CNN comes to solve this issue. The above diagram represents a deep neural network with taking input as all 3 million pixels values of the image, which further classify it into one of the two predefined output class.

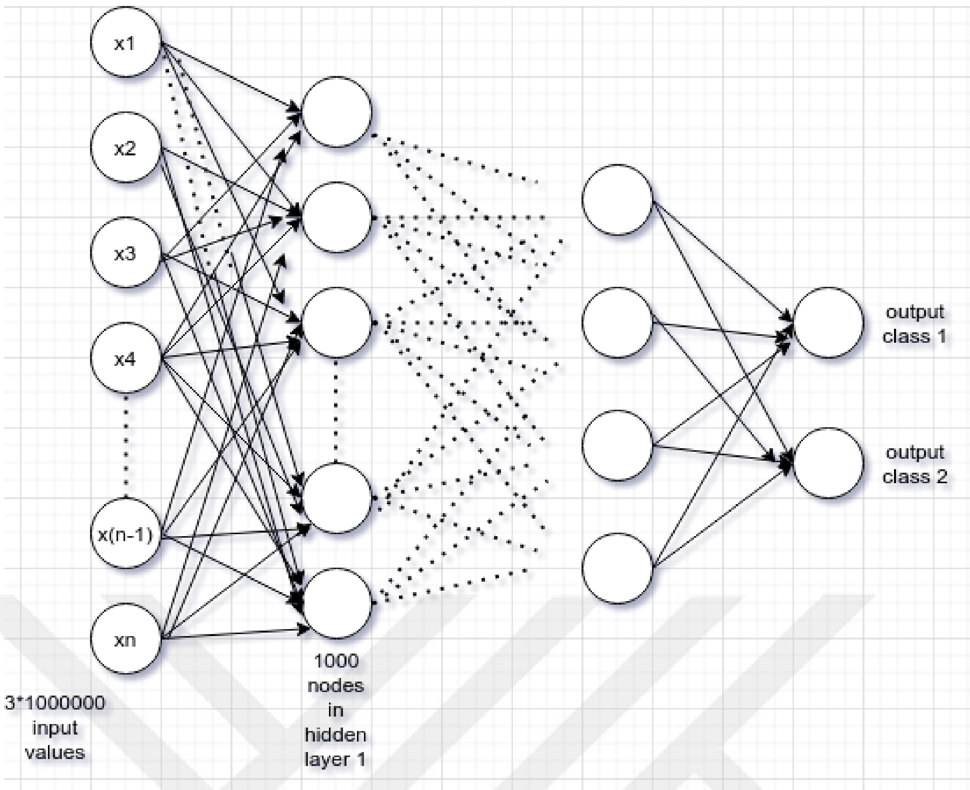


Figure 13: Deep neural network

3.4. Models

In order to predict the segmentation mask, four well-known models specialized for semantic segmentation were employed. These were the Convolutional Neural Networks (CNN), DNN, and YOLO. Their architecture and implementation will be explained in this section.

3.4.1. Artificial Neural Networks (ANN)

Artificial Neural Networks (ANN) are frameworks inspired by animal brains' biological natural neural systems. The neural network is not an algorithm in and of itself, but rather a structured framework for several Machine Learning algorithms to collaborate and handle complex information. Such frameworks "learn" to do tasks by seeing enough examples, without being coded to adhere to any explicit criteria. In image identification, for example, the artificial neural network is used to detect photographs of dogs by showing the network manually tagged images of dogs. This job does not require the network to have any prior knowledge of dogs, such as visual representations of them. A deep learning neural network is a type of artificial neural network architecture that has numerous layers between inputs and outputs. Shallow networks, on the other hand, have only one neural layer between inputs and outputs. Figure 14 depicts a comparison of two architectures.

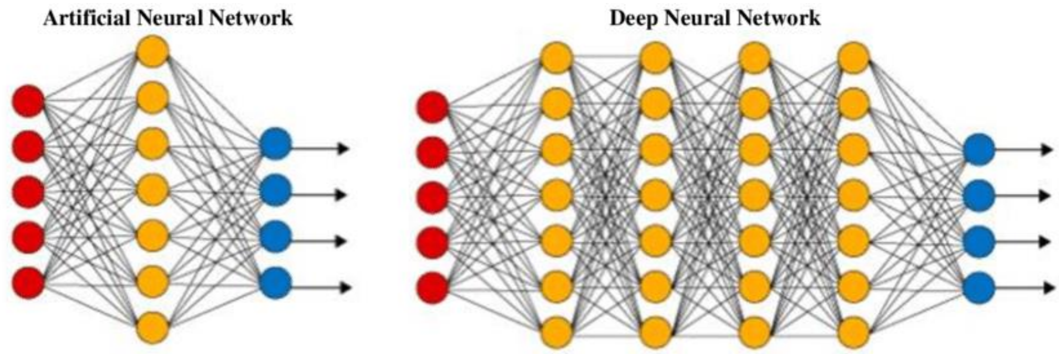


Figure 14: Comparison of two artificial neural network architectures

A collection of perceptrons and activation functions together form the artificial neural network (ANN). The connected perceptrons form hidden layers or units that map the input to output on non-linear basis.

3.4.2. CNN Model

Neither CNN nor any other model can be feasible to learn these many parameters, hence the basic fundamental structure of the CNN model is not like the deep neural networks but it works in a bit different way. The CNN utilizes a various leveled model to fabricate an organization, like a channel, and afterward yields a completely associated layer where every one of the neurons are associated and the result is handled. Every image contains some sort of feature dependencies at the pixel level which plays an important role in processing the image in the desired way and the convolutions operations in the CNN model act as image feature extractors automatically When a pixel vector algorithm is used, a lot of spatial interaction between pixels is lost, but adjacent pixel information is effectively used by a CNN to down sample the image first by convolution and then a prediction layer is used at the end. Since feature selection is done automatically by the CNN, there is no need to spend time on feature engineering before using it. Additionally, when the detected features of the CNN are compared to the handcrafted features, the performance is much better. Along with the global and local feature covering, CNN also learns a whole new set of features from the image itself.

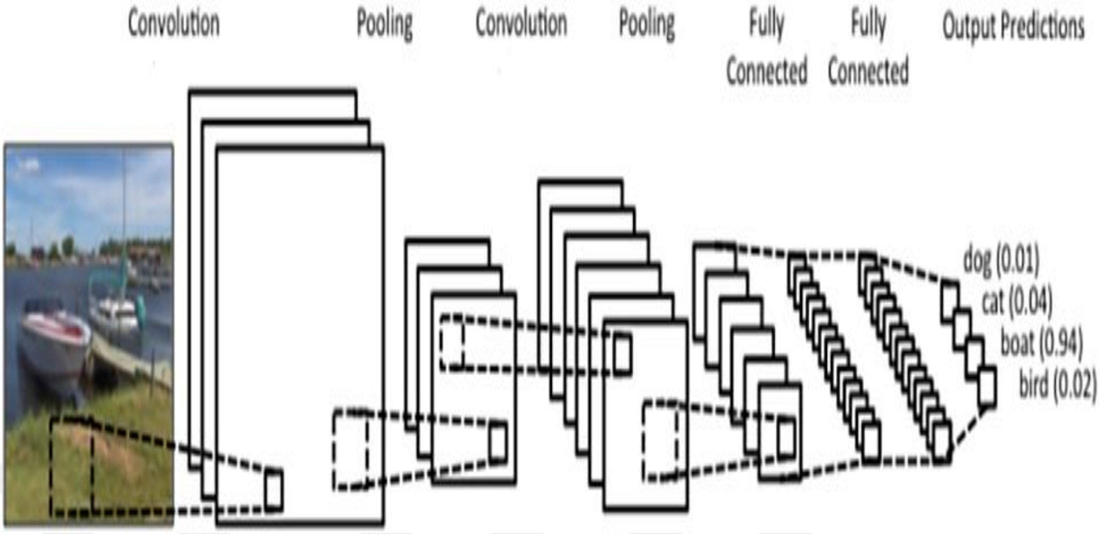


Figure 15: CNN Architecture

Every neuron in one layer is associated with all neurons in the accompanying layer. CNNs are less confounded in their regularized variant and have a few applications, for example, face or movement acknowledgment, text characterization, normal language handling, image grouping, and PC vision tasks. CNN utilizes a feed forward engineering.

3.4.3. You Only Look Once (YOLO-v5) Model

In this study YOLOv5 object detection algorithms are implemented which is released recently on 9 June 2020. Glenn Jocher, an unauthorized author, released YOLOv5 only four days after YOLOv4. There are several disagreements around the choice of the moniker "YOLOv5" and other issues. Glenn released a PyTorch-based version of YOLOv5 with significant enhancements. As a result, he has yet to issue an official statement. This version is rather impressive, outperforming all previous versions in terms of COCO AP and coming close to EfficientDet AP with greater FPS. For the generalization and accuracy purpose transfer learning is used to extract features and last few layer of model is retrained to get detection in our custom images. The purpose of YOLOv5 is not to achieve the best Map, but instead: • Ease of use • Exportability • Memory requirements • Speed • Map • Market size

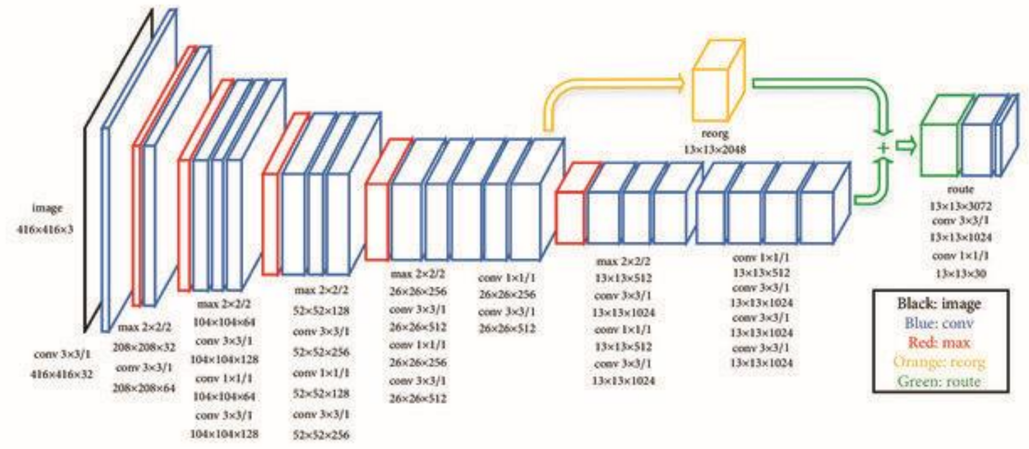


Figure 16: Proposed YOLO-v5 architecture

Our object detection deep neural network is implemented in pytorch on google colab GPU. The following steps are taken to train the detector: • Install YOLO-v5 dependencies • Preparation of dataset for YOLO-v5 Object Detection. Label is converted into 'text' format form 'xml' format. • Define YOLO-v5 Model Configuration and Architecture • Train a custom YOLO-v5 Detector • Evaluate YOLO-v5 performance • Visualize YOLO-v5 training data • Run YOLO-v5 Inference on test images • Export Saved YOLO-v5 Weights for Future Inference.

3.5. Performance Metrics

Performance measures may be used to evaluate the model's performance and efficiency. Additionally, several studies have used performance indicators to compare two or more models. It is crucial to choose the metrics properly since it increases the value of the research. Because model performance must be evaluated, the metrics used in this thesis to evaluate the models are Accuracy, Precision, Recall, and Training time. The following measures are also used to compare the models.

CHAPTER FOUR

RESULT AND DISCUSSION

The ideal option is to perform an experiment because the data utilized to attain the thesis's goals and objectives is quantitative in nature (optical and aerial images).

4.1. Preliminary

To examine all the models under the same circumstances, a parent class was implemented in which the training and validation of the models were defined, leaving only the segmentation model and related properties as abstract fields. Almost all models, except Mask-RCNN, were defined in classes that extended this parent class. This approach was feasible by implementing all models under the same backend, which is TensorFlow and Keras. In the parent class, the optimizer, loss function, and metrics were defined, as well as some callbacks to improve performance, the number of epochs, batch size, and the way in which their performance was evaluated during testing. ADAM optimizer was chosen as it is a very efficient optimizer that resembles RMSProp with momentum. The learning rate was set to 1e−3 and periodically decreased, with the weight decay set to 1e−6. A scheduler was set using callbacks to halve the learning rate every five epochs and reduce the LR by a small factor when in a plateau. Figure 4.9 displays the scheduled LR values during training. Another callback utilized was the checkpoint callback, which stored the weight of the models each time a local minimum (i.e. the metric was maximized, or the loss was minimized) was detected, along with early stopping. These callbacks were applied to the model during the training process.

For metrics, the DICE coefficient and intersection over union (IoU) were utilized, which are widely used in semantic segmentation. The DICE coefficient measures the similarity between two sets A and B and is defined as DICE = 2AB/(A+B). Similarly, IoU is another similarity function that measures overlap between two objects and is defined as IoU = (AB)/(A+B). It is also frequently used in object detection. As for the loss function, binary cross-entropy subtracted by the DICE coefficient was employed. It has been noted in forums that using DICE or IoU as loss functions does not produce good gradients, leading to slow and unstable training.

Many similar projects have suggested a combination of DICE and cross entropy as a solution.

The code pertaining to CNN and ANN involves loading the model, configuring its hyperparameters, and extending its data loader class. The same callbacks were passed to improve the performance. In terms of loss functions, CNN utilizes a loss function that is a sum of smaller loss functions.

Table 2: The performance of the Models, regarding DICE coefficient and IoU, over the testing set

Model	DICE	IoU
ANN	0.35	0.21
CNN	0.51	0.39

4.2. Experimental Setup

All new advances in the field of AI and deep learning have been achieved using Python libraries. Accordingly, it is generally used as the significant programming language for making a wide range of calculations, projects, and scripts (Ren, He, Girshick, & Sun, 2019). Jupyter Note pad is a refined open-source application that permits designers to make and impart reports to live code, conditions, perception, and text. It tends to be utilized for different applications, for example, information handling, information investigation, AI, measurable displaying, and numerous others.

The details of the proposed method for tweaking the hyperparameters of the ANN and CNN algorithms were depicted in this section. The CNN strategy was picked for this study in light of the fact that different scientists in the articles remembered for the writing survey utilized it and discovered that it was helpful. As per the writing audit examination, CNN furnishes high precision with high location speeds, though the DNN calculation furnishes exceptionally high discovery speeds with continuous execution yet experiences exactness. CNN, since it is in the middle and delivers the best of both worlds, can achieve good results. The evolutionary algorithms are used in this work to tune the hyperparameters. The employment of genetic algorithms has been dubbed an evolutionary deep learning technique. The proposed technique for detecting ships is based on the CNN and ANN algorithms.

4.3. Ship Detection Dataset Parameters

The assortment comprises of images of boats removed from Planet satellite symbolism gathered over California's San Francisco and San Pedro Bayous. There are 4000 photos, 80x80, 3-channel design (RGB images), marked "transport" or "no-transport." Planet.com gave images of boats and full-outline visual items. By giving a shipsnet.zip compress design registry containing the entire dataset as.png transport images.

Each image filename in our dataset follows a specific naming convention consisting of several components. First, the label of the image is represented by a value of either one (1) or zero (0), corresponding to the "ship" or "noship" class, respectively. In addition, the filename includes a scene id that, when used in conjunction with the Planet application, can be used to uniquely identify the entire scene. Finally, the filename includes the longitude and latitude coordinates of the image, separated by an underscore, providing the exact location where the photo was taken. With this standardized naming convention, specific images can be easily located and analyzed with precision and accuracy.

Moreover, the dataset is organized in a JSON text record named shipsnet.json, which incorporates information, mark, scene_id, and area records. In the information, every pixel in the 4000 images is addressed by 19200 numbers. The red channel is the initial 6400 passages, trailed by the green channel, and ultimately by the blue channel. The image is put away straight situated succession, with the initial 80 components of the cluster conveying the image's most memorable line's red channel values. The ‘ship' class has 1000 photographs zeroed in on the body of a solitary boat of changing sizes and climatic circumstances. Figure 17 portrays many images from this class.

Table 3: SHIPSET Dataset Plot

	data	labels	locations	scene_ids
0	[82, 89, 91, 87, 89, 87, 86, 86, 86, 86, 84, 8...	1	[-118.2254694333423, 33.73803725920789]	20180708_180909_0f47
1	[76, 75, 67, 62, 68, 72, 73, 73, 68, 69, 69, 6...	1	[-122.33222866289329, 37.7491755586813]	20170705_180816_103e
2	[125, 127, 129, 130, 126, 125, 129, 133, 132, ...	1	[-118.14283073363218, 33.736016066914175]	20180712_211331_0f06
3	[102, 99, 113, 106, 96, 102, 105, 105, 103, 10...	1	[-122.34784341495181, 37.76648707436548]	20170609_180756_103a
4	[78, 76, 74, 78, 79, 79, 79, 82, 86, 85, 83, 8...	1	[-122.34852408322172, 37.75878462398653]	20170515_180653_1007
...	...	...	...	...
995	[103, 95, 99, 108, 110, 125, 131, 125, 133, 13...	1	[-122.34862085843947, 37.767167357603554]	20170921_181406_1031
996	[75, 75, 75, 75, 74, 70, 69, 72, 73, 72, 70, 6...	1	[-122.33786604607423, 37.73944613318674]	20170604_180820_0f52
997	[75, 74, 75, 75, 74, 74, 75, 76, 76, 75, 75, 7...	1	[-122.33284969739871, 37.71792145705744]	20170721_180825_100b
998	[49, 50, 49, 45, 49, 53, 52, 46, 43, 44, 49, 5...	1	[-122.35668820008198, 37.75991104734941]	20170618_180801_0f34
999	[113, 114, 112, 110, 111, 111, 114, 110, 112, ...	1	[-122.35032196580875, 37.767393062977916]	20170522_180635_0f42

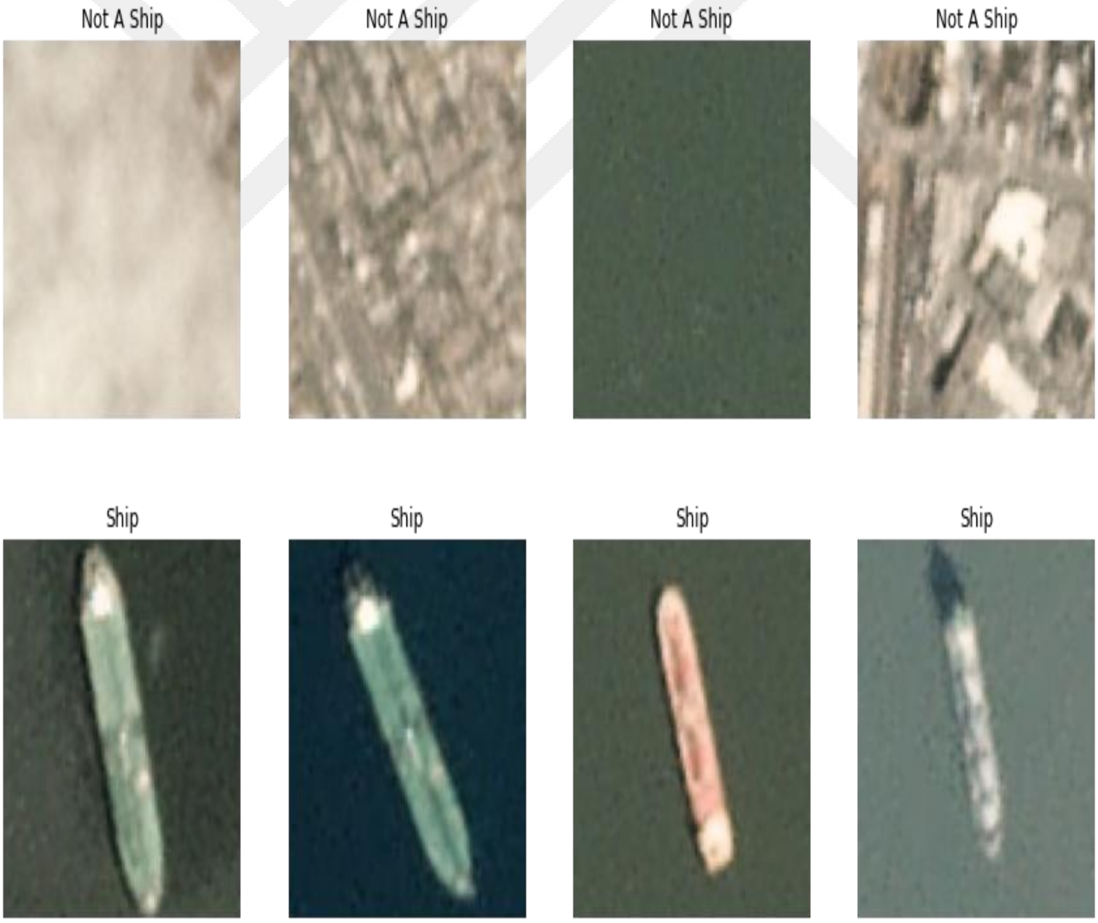


Figure 17: Exploring the images

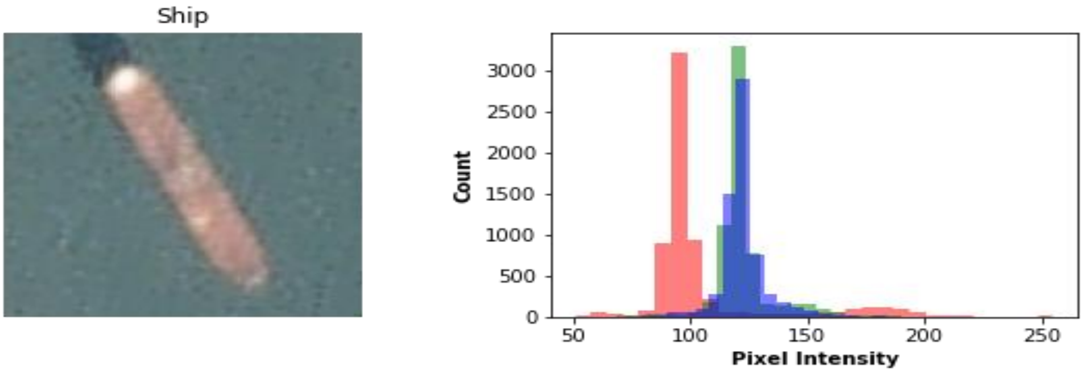


Figure 18: Image contains ship using Pixel Intensity

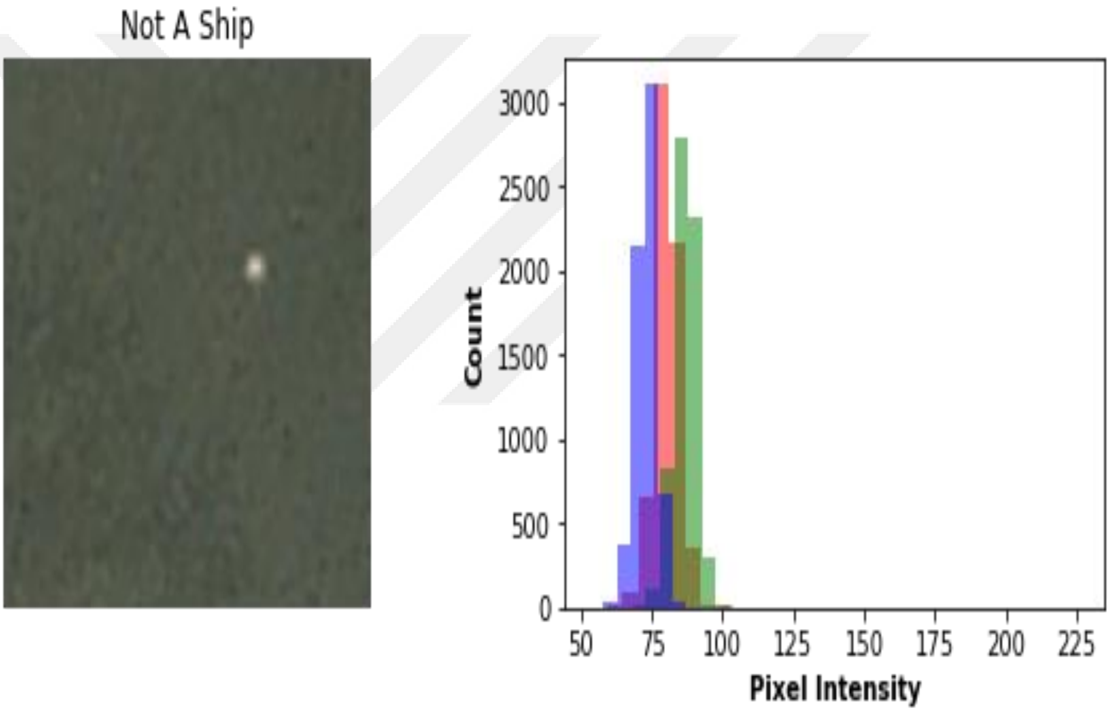


Figure 19: Image does not contain ship using Pixel Intensity

4.4. Performance

Before training, the training dataset is split into training and validation sets. The initial training set consists of around 4000 images, and around 800 images of them are used to form the validation se. The training function receives as input the generator that produces the augmented data, and trains for 22 epochs. The results are displayed in Table 3. The experiments were performed into my laptop containing an Intel i7 9th generation CPU, 8GB memory and a GTX1050 GPU. Experiments were attempted on the Collab TPU cluster11, but due to having to store the images in the Google Drive and the resulting deterioration in the network I/O for reading the images, any advantage of the TPUs was erased. As you can see, none of the models manage to achieve high DICE or IoU over the testing set, however all of the models seem to be able to detect the ships but the predicted masks lack precision. In most cases the ships are very small and hence their masks are very small, while the predicted masks are blobby and lack boundary details. As can be seen in Figure 4.4, the loss function has not converged after 22 epochs, indicating that better results can be achieved with further training. The CNN, being a big pre-trained model, probably requires more training. Based on the observation of its loss during training in Figure 20, it can be reckoned that the model has not converged after 22 epochs and requires further training, with an even smaller learning rate. It was believed that using gradient descent with momentum would improve the performance of the CNN, but CNN has its own optimization technique that is not open to change.

4.4.1. ANN Performance

For training implementation, the ANN architecture was updated. The network's output layer was eliminated. A FC layer of 512 neurons that are activated by ReLU. Softmax activation was used to connect the FC layer to the output layer. For improved training regularization, dropout connections with a 25% rate were added after the FC layer. The revised model had 24.6 million parameters. Two models with learning rates of 103 and 104 were trained. The latter achieved higher validation accuracy. Figures 4.5 and 4.6 depict the training procedure for the superior model. In comparison to the previous two models, the model trained for a maximum of 50 epochs and no early stopping requirement was met. It achieved a training accuracy of 94% and validation accuracy of 93%.

Basic imports are utilized before training anything. TensorFlow is a machine learning package with a strong emphasis on neural networks. Pandas is a data-parsing library in the form of a spreadsheet. Finally, NumPy speeds up and simplifies number crunching. The pixel values are saved in the data frame in a column called "data." These pixel values aren't ready to be processed by a CNN as is. To normalize the values, the new data is converted to a NumPy array and divided by 255. All of the

19,200 values should now be between 0 and 1. The data is then molded to an 80 × 80 x 3 matrix and formatted as an image.

```
Epoch 4/100
75/75 [==============================] - 7s 96ms/step - loss: 0.3507 - accuracy: 0.8625 - val_loss: 0.2875 - val_accuracy: 0.88
38
Epoch 5/100
75/75 [==============================] - 7s 99ms/step - loss: 0.3159 - accuracy: 0.8692 - val_loss: 0.2959 - val_accuracy: 0.88
75
Epoch 6/100
75/75 [==============================] - 7s 95ms/step - loss: 0.2962 - accuracy: 0.8833 - val_loss: 0.2574 - val_accuracy: 0.89
50
Epoch 7/100
75/75 [==============================] - 7s 96ms/step - loss: 0.2897 - accuracy: 0.8813 - val_loss: 0.2425 - val_accuracy: 0.89
75
Epoch 8/100
75/75 [==============================] - 7s 96ms/step - loss: 0.2527 - accuracy: 0.8946 - val_loss: 0.2546 - val_accuracy: 0.90
75
Epoch 9/100
75/75 [==============================] - 7s 97ms/step - loss: 0.2987 - accuracy: 0.8829 - val_loss: 0.2657 - val_accuracy: 0.88
75
Epoch 10/100
75/75 [==============================] - 7s 95ms/step - loss: 0.2419 - accuracy: 0.9033 - val_loss: 0.2333 - val_accuracy: 0.91
75
Epoch 11/100
75/75 [==============================] - 7s 96ms/step - loss: 0.2421 - accuracy: 0.9021 - val_loss: 0.3274 - val_accuracy: 0.86
12
Epoch 12/100
75/75 [==============================] - 7s 97ms/step - loss: 0.2482 - accuracy: 0.9129 - val_loss: 0.1948 - val_accuracy: 0.92
12
Epoch 13/100
75/75 [==============================] - 7s 98ms/step - loss: 0.2372 - accuracy: 0.9079 - val_loss: 0.2147 - val_accuracy: 0.91
62
Epoch 14/100
75/75 [==============================] - 7s 99ms/step - loss: 0.2172 - accuracy: 0.9146 - val_loss: 0.2234 - val_accuracy: 0.91
12
Epoch 15/100
75/75 [==============================] - 7s 98ms/step - loss: 0.2501 - accuracy: 0.9083 - val_loss: 0.1950 - val_accuracy: 0.91
62
Epoch 16/100
75/75 [==============================] - 7s 99ms/step - loss: 0.2006 - accuracy: 0.9187 - val_loss: 0.2060 - val_accuracy: 0.92
50
Epoch 17/100
75/75 [==============================] - 7s 99ms/step - loss: 0.1893 - accuracy: 0.9271 - val_loss: 0.4375 - val_accuracy: 0.81
12
Epoch 18/100
75/75 [==============================] - 7s 99ms/step - loss: 0.2330 - accuracy: 0.9058 - val_loss: 0.2028 - val_accuracy: 0.91
00
Epoch 19/100
75/75 [==============================] - 7s 100ms/step - loss: 0.1779 - accuracy: 0.9287 - val_loss: 0.2260 - val_accuracy: 0.9
200
Epoch 20/100
75/75 [==============================] - 7s 98ms/step - loss: 0.1910 - accuracy: 0.9262 - val_loss: 0.3056 - val_accuracy: 0.87
37
Epoch 21/100
75/75 [==============================] - 7s 99ms/step - loss: 0.1583 - accuracy: 0.9379 - val_loss: 0.2072 - val_accuracy: 0.93
62
Epoch 22/100
75/75 [==============================] - 8s 101ms/step - loss: 0.1640 - accuracy: 0.9358 - val_loss: 0.2278 - val_accuracy: 0.9
312
```

Figure 20: ANN Compilation & Prediction Results of Ships Detection Application

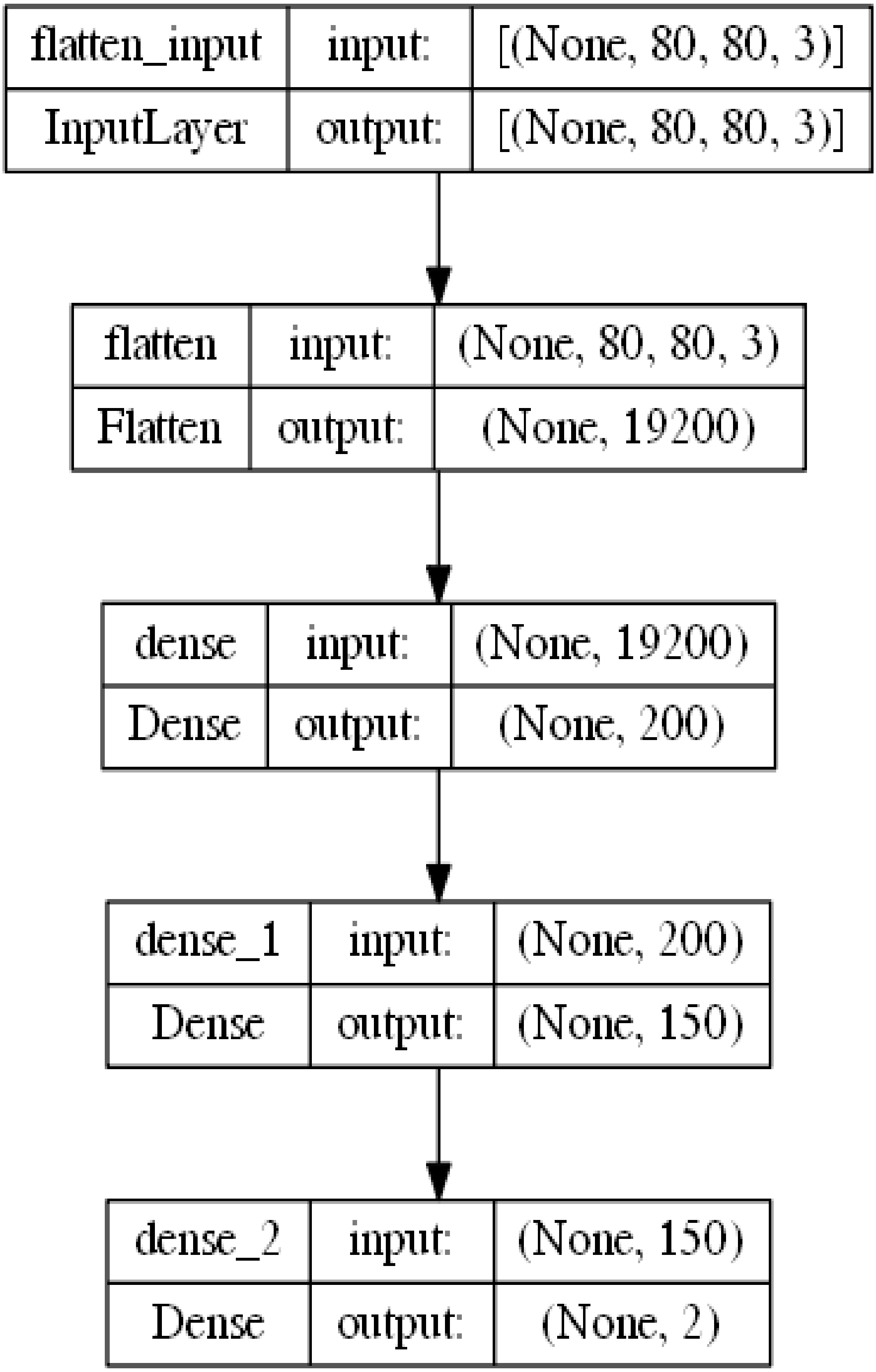
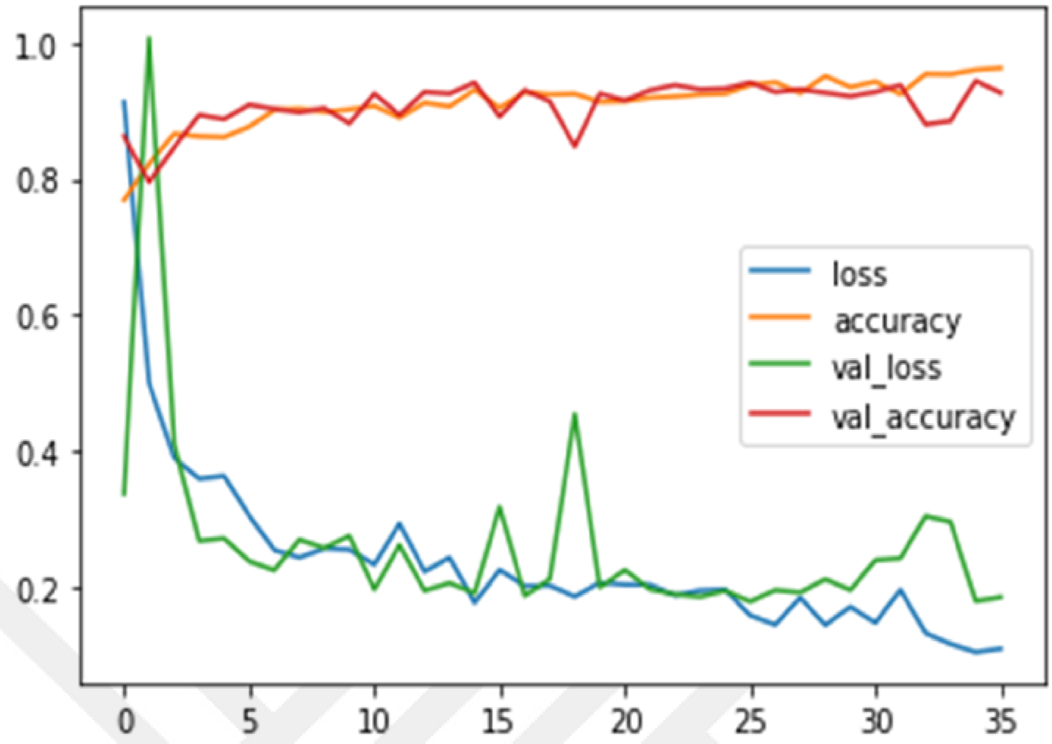
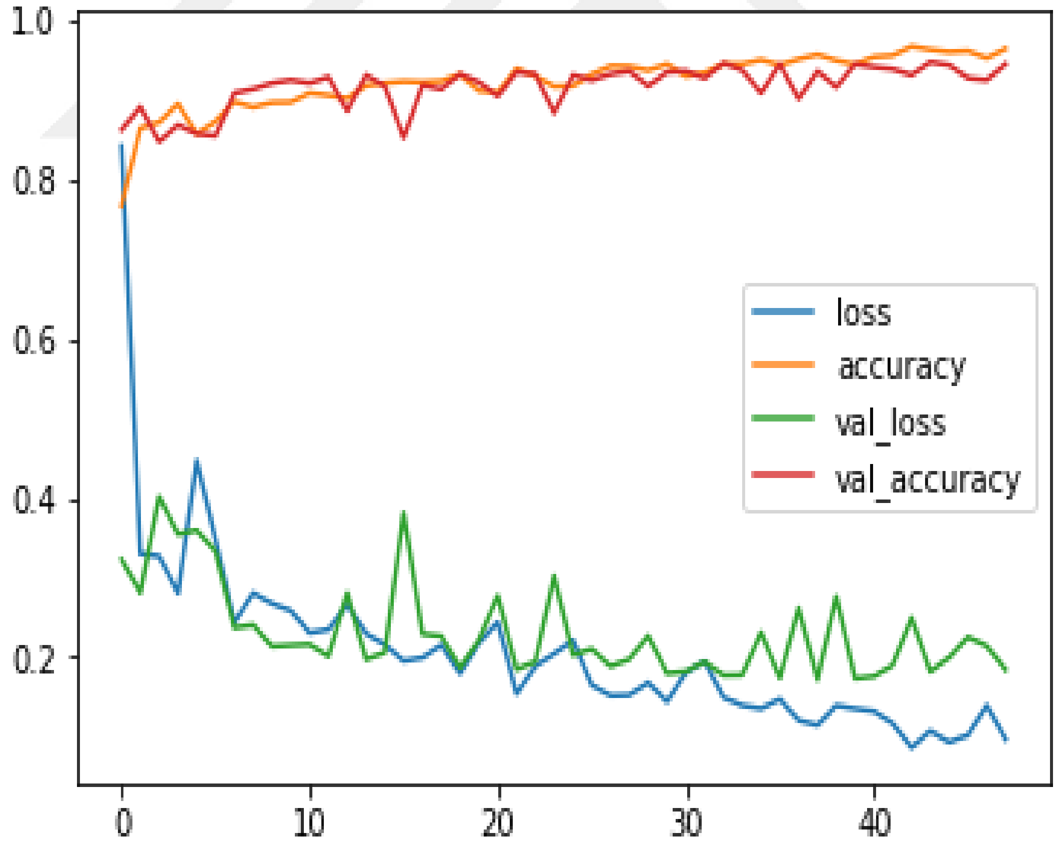


Figure 21: representation of model layers



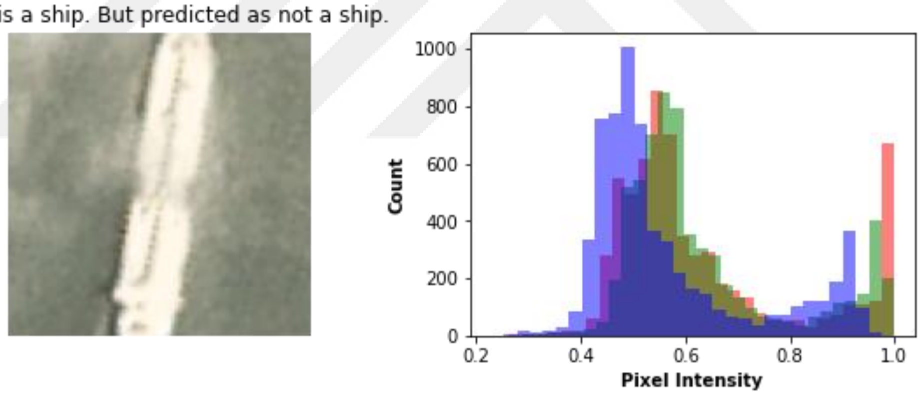


Graphic 1. ANN architecture Accuracy results

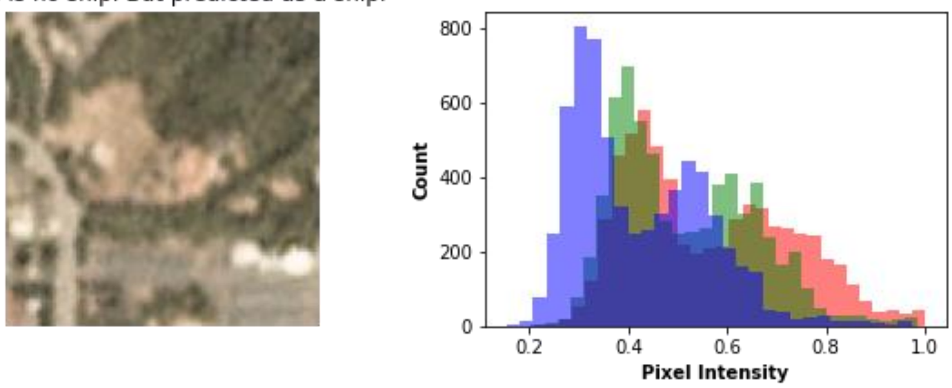
Table 4: ANN predicted data results

	Not A Ship	Ship	There is a Ship	Difference
0	0.396110	0.800216	1.0	-0.199784
1	0.889800	0.123727	0.0	0.123727
2	0.215726	0.880204	1.0	-0.119796
3	0.391349	0.874571	0.0	0.874571
4	0.917596	0.001272	0.0	0.001272
...	...	...	...	...
705	0.584181	0.678428	0.0	0.678428
706	0.439273	0.812180	1.0	-0.187820
707	0.821926	0.297751	0.0	0.297751
708	0.933961	0.142170	0.0	0.142170
709	0.920424	0.211472	0.0	0.211472

There is a ship. But predicted as not a ship.



There is no ship. But predicted as a ship.



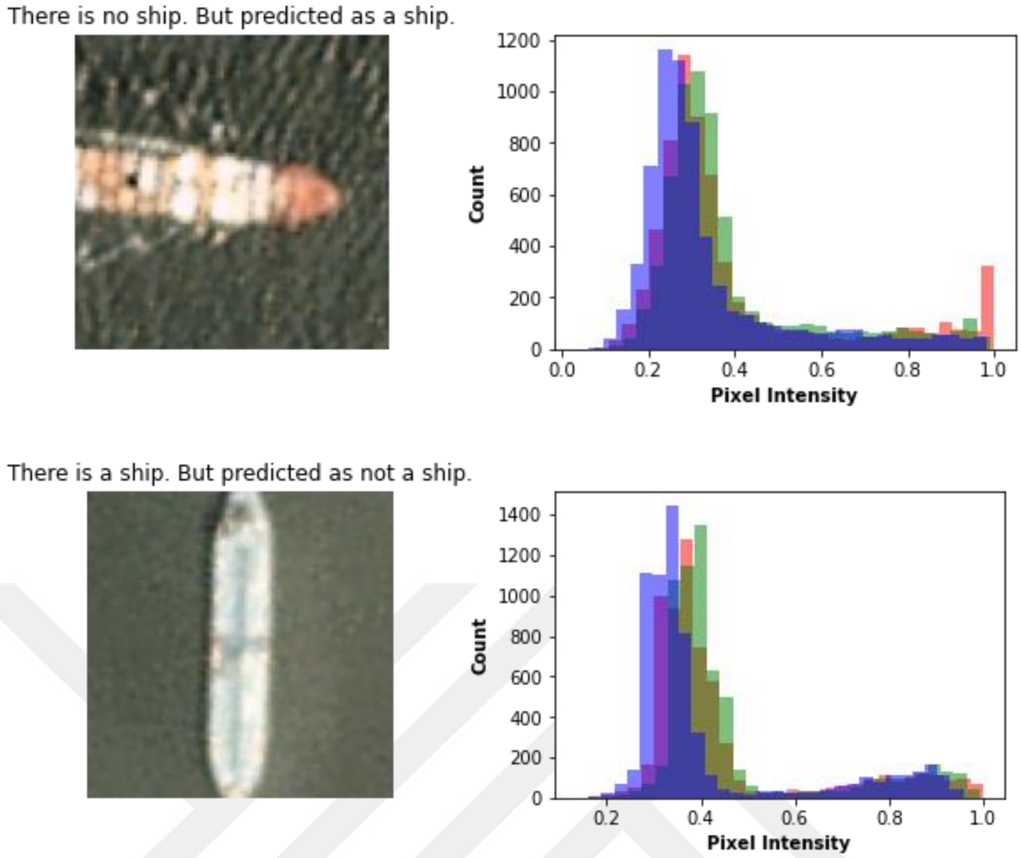


Figure 22: ANN image classification results

4.4.2. CNN Performance

Graph 2: depicts the data obtained when the experiments were completed. The evolutionary algorithm tuned the network parameters at 0.0035 learning rate and 4 batch size. The exactness rates showed by CNN are 96% and 97%, separately. As far as precision, the new strategy beats the first organization structure. Albeit the exactness is only one rate point more prominent than the first, the model's accuracy in precisely perceiving ships is likewise higher.

Following the execution of the picking up/preparing process, the model on ships satellite images, utilizing the pre-referenced boundaries, insofar as noticing the preparation exactness and misfortune capability results, inferred that the model result, for the recently referenced, number of ages, the training accuracy is 98,52% and the training loss is sufficient low, practically 4,38%, while testing accuracy is 99,14% and testing loss is 3,26%.

```
history = model.fit(datagen.flow(x_train, y_train), epochs = 100,
                    validation_data=(x_val, y_val), callbacks = [earlystopping])
```

```
Epoch 1/100
75/75 [==============================] - 9s 106ms/step - loss: 0.2100 - accuracy: 0.9254 - val_loss: 0.0816 - val_accuracy: 0.9
638
Epoch 2/100
75/75 [==============================] - 8s 104ms/step - loss: 0.1243 - accuracy: 0.9529 - val_loss: 0.1680 - val_accuracy: 0.9
237
Epoch 3/100
75/75 [==============================] - 8s 108ms/step - loss: 0.1366 - accuracy: 0.9496 - val_loss: 0.0700 - val_accuracy: 0.9
762
Epoch 4/100
75/75 [==============================] - 8s 104ms/step - loss: 0.0934 - accuracy: 0.9667 - val_loss: 0.0716 - val_accuracy: 0.9
737
Epoch 5/100
75/75 [==============================] - 8s 108ms/step - loss: 0.0907 - accuracy: 0.9683 - val_loss: 0.0643 - val_accuracy: 0.9
775
Epoch 6/100
75/75 [==============================] - 8s 109ms/step - loss: 0.1034 - accuracy: 0.9642 - val_loss: 0.0704 - val_accuracy: 0.9
787
Epoch 7/100
75/75 [==============================] - 9s 120ms/step - loss: 0.1082 - accuracy: 0.9604 - val_loss: 0.0977 - val_accuracy: 0.9
563
Epoch 8/100
75/75 [==============================] - 10s 136ms/step - loss: 0.0928 - accuracy: 0.9721 - val_loss: 0.0644 - val_accuracy: 0.
9775
Epoch 9/100
75/75 [==============================] - 11s 143ms/step - loss: 0.1207 - accuracy: 0.9621 - val_loss: 0.1490 - val_accuracy: 0.
9400
Epoch 10/100
75/75 [==============================] - 10s 136ms/step - loss: 0.1105 - accuracy: 0.9592 - val_loss: 0.0786 - val_accuracy: 0.
9737
Epoch 11/100
```

Figure 23: CNN Compilation & Prediction Results of Ships Detection Application

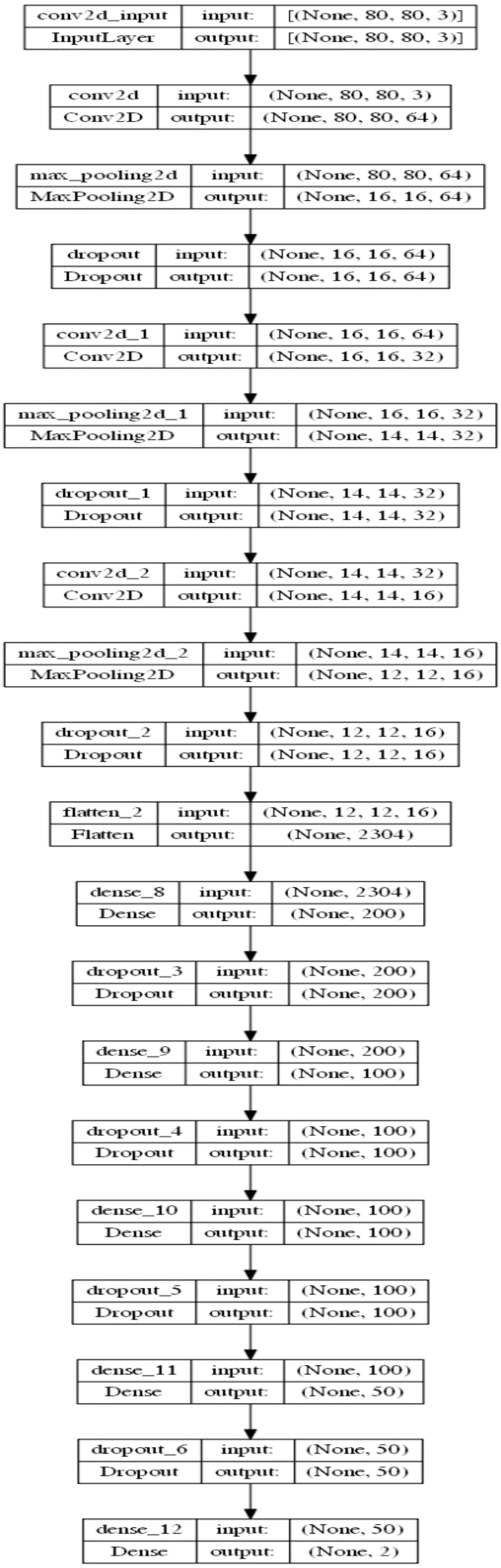
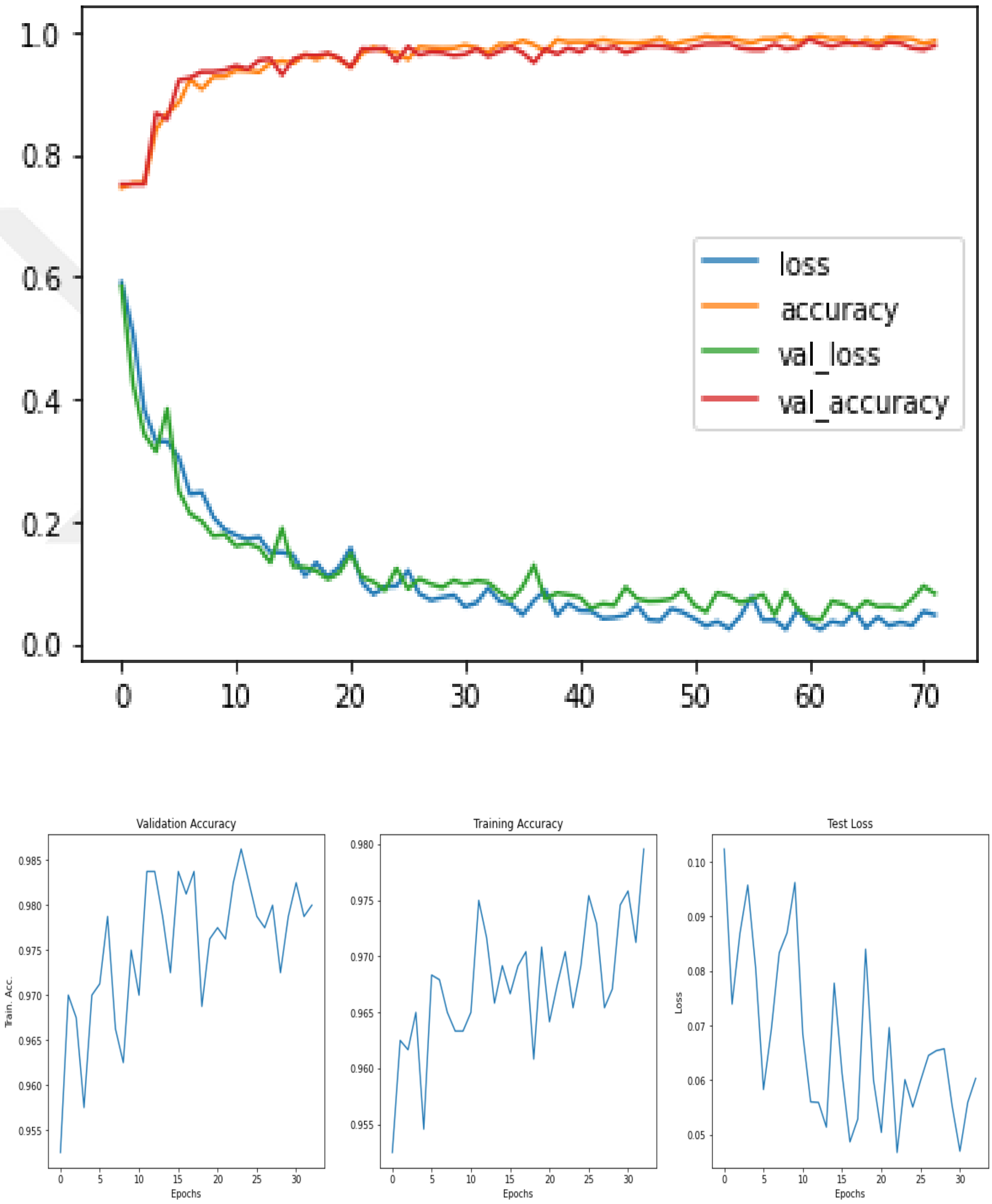


Figure 24: Representation of model layers

When evaluating the model's performance on the test set, it is clear that both validation loss and accuracy are in harmonic synchronization with the corresponding training features, implying that the model was not over-fitted. It was determined that the model's efficiency increased when the loss test process was almost similar to training loss, as the first (validation) is lowered throughout the epochs, while the gap between training and validation accuracy was erased.



Graphic 2: Accuracy and Loss Plots Between Training and Validation Data

The testing data included 1000 photos, 733 of which were non-ships and 267 of which were ships. The precision for non-ships is 99%, which is somewhat higher than the 97% for ships. Essentially, 97% of the photos categorized as ships by the model were actual ships. The model correctly identified 99% of non-ship photos and 98% of non-ship images, respectively

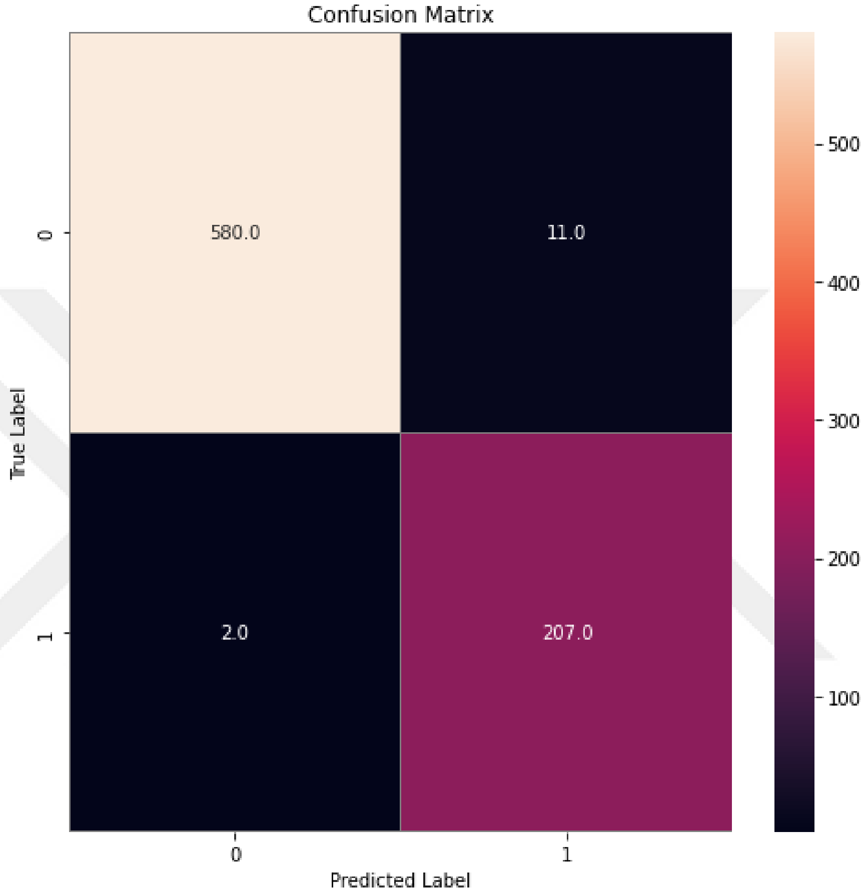


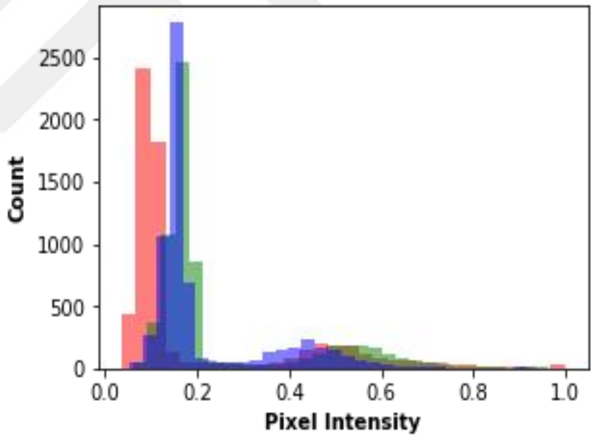
Figure 25: CNN normalized confusion matrix

Exactness and Misfortune Plots (Figure 25), as well as Expectations results, are saved in HDFS in the 'Expectations Loads' registry to be accessible whenever another forecast cycle is required, without the necessity for model retraining. Keras models saved in HDF5 (.h5) design - ideal for putting away multi-faceted varieties of numbers-- save days, in the event that not long stretches of model (re)training time. The last step after the preparation methodology is to assess the neural network. In this situation, an endeavor was made to get a feeling of how well the model performed by choosing 10 irregular photographs from approval information and obtaining the expected result as a name.

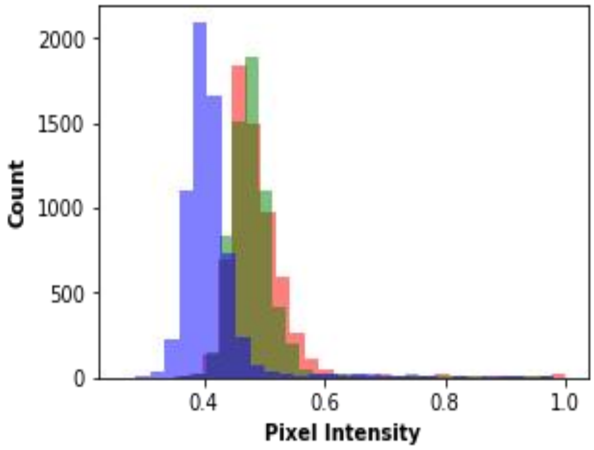
Table 5: The table shows the results of ship detection

	Not A Ship	Ship	There is a Ship	Difference
717	0.078492	0.921507	0.0	0.921507
699	0.094824	0.905176	0.0	0.905176
747	0.158780	0.841220	0.0	0.841220
461	0.327896	0.672104	0.0	0.672104
32	0.337117	0.662883	0.0	0.662883
217	0.385741	0.614259	0.0	0.614259
274	0.391165	0.608835	0.0	0.608835
136	0.429465	0.570535	0.0	0.570535
526	0.447985	0.552015	0.0	0.552015
564	0.459198	0.540802	0.0	0.540802

There is no ship. But predicted as a ship.



There is no ship. But predicted as a ship.



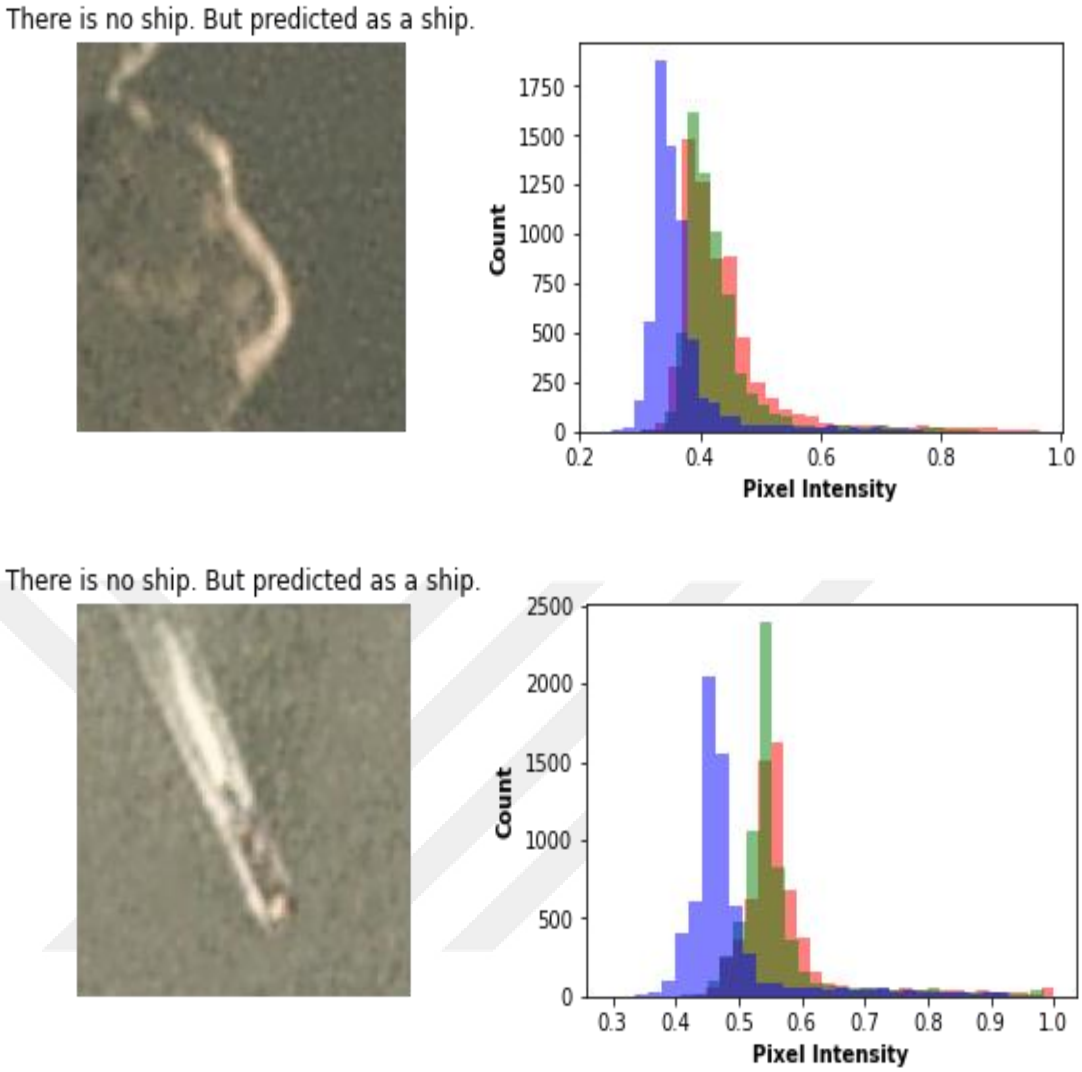


Figure 26: Representation of model layers

4.4.3. YOLO-v5 Model Performance

The model was trained for 1500 iterations and various observations were plotted in Figure 4.11. It was found that precision, recall, and MAP improved with respect to epochs and went into saturation at around 1500 iterations, indicating no need for further training. The model was also tested on validation data, and samples of YOLOv5's detection capability were shown in Figure 4.11

Figure 27: Images of test data showing the detection results

This shows that our model is capable of detecting various objects with much confidence. It also yields an acceptable accuracy and can be implemented in real time as it takes about 21ms/per image in detection of multiple objects associated with that image., which means it can be executed rapidly and implemented online and that its performance is competitive with current popular methods. Because of its prominent learning capability, it avoids the problems associated with the effects of environmental illumination changes.

An accuracy of 95% was obtained using the test data, this is because a convolutional network and dropout were used to avoid overfitting, a simplification of the YOLOV2 architecture was used, this type of network is efficient in classification problems. of images. Something that can be done to improve the model is to implement data argumentation since there are only 2500 images in total.

4.5. Comparaison

Based on the literature review, one of the three best-performing algorithms, namely the DNN and CNN algorithms, is selected as the preferred strategy for carrying out the experiment. The DNN and CNN algorithms with genetic algorithms were applied on all three datasets relating to aviation, vehicles, and ships. The models are evaluated using the previously indicated performance measures. This section offers comprehensive results from all tests and models.

Table 6 presentations the exactness, accuracy, and review for transport recognition. Exactness, accuracy, and review were determined utilizing execution measures. The hereditary calculation gave values to the learning rate and bunch size of 0.0035 and 8, individually. CNN's exactness with a hereditary not entirely settled to be 4% more prominent than that of the DNN approach. The ideal YOLOv3 not entirely settled to be 81.36% and 85.41%. Moreover, the precision values are fairly more noteworthy. Both the first and redesigned models accomplished remarkable accuracy of 0.88 and 0.91, individually. It is commonly realized that accuracy and review are connected, and that as precision improves, the review esteem drops. The experiment findings show that the original and upgraded models have recall values of 0.717 and 0.694, respectively.

Table 6 thinks about the consequences of YOLOv3 and YOLOv4 with proposed model methods for recognizing ships from satellite information. The transformative calculation confirms that the ideal hyperparameters for identifying ships are 0.004 and 8 for the learning rate and clump size. The precision values for the first and further developed models are 83% and 86%, separately. Also, the accuracy values for the ordinary and improved models are 0.847 and 0.824, separately. For the two models, review values are determined to be 0.68.

Table 6: The table shows the results of ship detection j

Algorithms	Accuracy	Precision	Recall
Proposed DNN	93.28%	93%	94%
Proposed CNN	97.14%	99%	97%
Proposed YOLO-v5	98.50%	93%	95%
YOLOv4 (Cordova A. W., Quispe, Inca, Choquehuayta, & Gutierrez, 2020)	93.57%	84.7%	68%
YOLOv3 (Thoudoju & Kusetoğulları, 2021)	86.78%	82.4%	68.8%

Table 6 showcases the aftereffects of the calculations used to perceive ships in satellite symbolism. The developmental calculation confirms that the ideal hyperparameters for identifying ships are 0.004 and 8 for the learning rate and clump size. The exactness values for the first and further developed models are 83% and 86%, individually. Likewise, the accuracy values for the customary and upgraded models are 0.847 and 0.824, separately. For the two models, review values are determined to be 0.68.

The results of the experiments reveal that accuracy for CNN performs better than other models for all of the ship detection performed in this thesis. The explanation for the increased accuracy could be that the genetic algorithm returns values for learning rate that are higher than the default network settings.

CONCLUSION AND FUTURE WORKS

The consequences of the writing survey and the examinations acted in the review are all given in Section 5

Conclusion

In this project, state-of-the-art models were employed to predict the segmentation masks of satellite images and detect ships. Fine results were produced by all the models, regardless of the relatively low DICE coefficient. Additionally, performance can be improved by applying filters that enhance the contrast of images. If one of these models had to be chosen, either CNN, ANN and YOLO-v5 combined with Adaptive equalizer, would be chosen as it produces very good results and is faster than the rest in both training and inference.

However, it is believed that the main reason for the bad performance is the downscaling of the images, considering that most of the ships are formed by just a few pixels. Down-scaling was, however, necessary to enable training on the machine used. For future work, to improve performance, the images would not be down-scaled, and more images would be used for training. The original dataset comprises approximately 200K images, while only 4K of them were used. Additionally, training for more epochs with an appropriate learning rate schedule would probably boost performance.

Additionally, it would be of interest to examine the performance of other models. DeepLabV3, another famous model used for semantic segmentation, was attempted to be integrated into the code, but failed. Another model to be considered is Random Walk Network, which produces fine-detailed segmentation masks, but its source code is not yet available. Incorporating Deep learning and move learning is incredibly valuable for growing limited scope models, such as the ones created by this proposal. The proposition yield produced as the model can be utilized to screen ships in the future to get familiar with their example and development practices, as well as being utilized on past images to find out about how ships were utilized beforehand. Thus, these procedures can help with accomplishing the embodiment of sea observing and reconnaissance, which incorporates checking ships for safe route, forestalling criminal operations, keeping up with line security, and safeguarding marine natural and organic biological systems from illicit fishing, oil slicks, and contamination.

Future Works

Even though time and assets were restricted, this proposition has opened the entryway for future investigation into growing new applications for the oceanic area in an assortment of ways:

• This postulation comprises exclusively of image grouping and acknowledgment concerning transport classes. It doesn't pinpoint the area of the boat in an image.

• The field of Deep learning is changing rapidly to the point that the structures utilized, for example, tensorflow, Keras, and related bundles, are being refreshed and refined consistently to give less intricacy as far as programming calculations and more prominent computational effectiveness. Staying up with the latest with progressing fast mechanical advancements is accordingly one more test for all Deep learning research works.

REFERENCES

Alghazo, J., Bashar, A., Latif, G., & Zikria, M. (2021). Maritime Ship Detection using Convolutional Neural Networks from Satellite Images. *10th IEEE International Conference on Communication Systems and Network Technologies. 74*, s. 21079214. Bhopal, India: IEEE Xplore. doi:https://doi.org/10.1109/CSNT51715.2021.9509628

Alibrahim, H., & Ludwig, S. A. (2021). Hyperparameter Optimization: Comparing Genetic Algorithm against Grid Search and Bayesian Optimization. (s. 978-1-7281-8393-0/2). 1 IEEE Congress on Evolutionary Computation (CEC). doi:https://ieeexplore.ieee.org/abstract/document/9504761/?casa_token=asC0V9jU2m0AAAAA:At3yS1Yl8W3DSELE_fLonZEZhub0fDHEsyhH3vwm7LkJdeAsu3Uix5Ozr2j7z3RrAT8YRgR4Mnk

Alzubi1,, J., Nayyar, A., & Kumar, A. (2018). Machine Learning from Theory to Algorithms: An Overview. *NCCI 2018. 1742-6596*, s. 1142. Journal of Physics. doi:https://iopscience.iop.org/article/10.1088/1742-6596/1142/1/012012

Ammar, A., Koubaa, A., Ahmed, M., Saad, A., & Benjdira, B. (tarih yok). Aerial Images Processing for Car Detection using Convolutional Neural Networks: Comparison between Faster R-CNN and YoloV3. *Journal Not Specified 2021, 1, 1*, s. https://doi.org/10.3390/1010001.

Aszemi, N. M., & Dominic2, P. .. (2019). Hyperparameter Optimization in Convolutional Neural Network using Genetic Algorithms. *(IJACSA) International Journal of Advanced Computer Science and Applications*. doi:https://pdfs.semanticscholar.org/c02f/877d81f487106cbd437f3f8d46b1496a897f.pdf

Bochinski, E., Senst, T., & Sikora, T. (2017). HYPER-PARAMETER OPTIMIZATION FOR CONVOLUTIONAL NEURAL NETWORK COMMITTEES BASED ON EVOLUTIONARY ALGORITHMS. (s. 2381-8549). IEEE. doi:https://doi.org/10.1109/ICIP.2017.8297018

Bradski, G., & Kaehler, A. (2008). Learning OpenCV Computer vision with the OpenCV library. doi:https://books.google.com.tr/books?hl=en&lr=&id=seAgiOfu2EIC&oi=fnd&pg=PR3&dq=The+OpenCV+Library&ots=hVM06ghHLd&sig=uVIZmAy8HG6jXy_lVHrCDXiwbIA&redir_esc=y#v=onepage&q=The%20OpenCV%20Library&f=false

Carlet, J., & Abayowa, B. (2017, sep 25). Fast Vehicle Detection in Aerial Imagery. doi:https://doi.org/10.48550/arXiv.1709.08666

Chen, X., Xiang, S., -Lin Liu,, C., & -Hong Pan, C. (2014). Aircraft Detection by Deep Belief Nets. *2013 Second IAPR Asian Conference on Pattern Recognition* (s. 0730-6512). IEEE. doi:https://doi.org/10.1109/ACPR.2013.5

Deng, Z., Sun, H., Zhou, S., Zhao, J., Lei, L., & Zou, H. (2017). Multi-scale object detection in remote sensing imagery with convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*. doi:https://www.sciencedirect.com/journal/isprs-journal-of-photogrammetry-and-remote-sensing

Domhan, T., Springenberg, J. T., & Hutter, F. (2015). Speeding Up Automatic Hyperparameter Optimization of Deep Neural Networks by Extrapolation of Learning Curves. *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence.* doi:https://ml.informatik.uni-freiburg.de/wp-content/uploads/papers/15-IJCAI-Extrapolation_of_Learning_Curves.pdf

Evans, B., Al-Sahaf, H., Xue, B., & Zhang, M. (2018). Evolutionary Deep Learning: A Genetic Programming Approach to Image Classification. doi:https://doi.org/10.1109/CEC.2018.8477933

FELZENSZWALB, P. F., & HUTTENLOCHER, D. P. (2003). Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision 59(2), 167–181, 2004*, 167–181, 2004. doi:https://link.springer.com/article/10.1023/B:VISI.0000022288.19776.77

Feng, X., Zhao, J., & Kita, E. (tarih yok). Genetic Algorithm based Optimization of Deep Neural Network Ensemble for Personal Identification in Pedestrians Behaviors. *2019 International Conference on Data Mining Workshops (ICDMW).* IEEE. doi:https://doi.org/10.1109/ICDMW.2019.00054

Fujino, S., Hatanaka, T., Mori, N., & Matsumoto, K. (2019). Evolutionary deep learning based on deep convolutional neural network for anime storyboard recognition. *Neurocomputing, 338,* 381-392. doi:https://doi.org/10.1016/j.neucom.2018.05.124

Gallego, A. -J., Pertusa, A., & Gil, P. (2018). Automatic Ship Classification from Optical Aerial Images with Convolutional Neural Networks. *remote sensing*. doi:https://doi.org/10.3390/rs10040511

Guo, J., He, H., Lausen, L., Li, M., Lin, H., Shi, X., . . . Xie, J. (2020). GluonCV and GluonNLP: Deep Learning in Computer Vision and Natural Language Processing. *Journal of Machine Learning Research 21 (2020) 1-7*. doi:http://jmlr.org/papers/v21/19-429.html

Ian, G., Yoshua, B., & Aaron, C. (2016). Book Review: Deep Learning., (s. 800). doi:https://doi.org/10.4258/hir.2016.22.4.351

KLUYVER, T., RAGAN-KELLEY, B., PÉREZ, F., GRANGER, B., BUSSONNIER, M., & FREDERICd, J. (2016). Jupyter Notebooks—a publishing format for reproducible computational workflows. doi:https://escholarship.org/content/qt08b3d4s2/qt08b3d4s2.pdf

Kumar, D., Zhang, X., Su, H., & Wei, S. (2022). Accurate Object Detection Based on Faster R-CNN in Remote Sensing Imagery. (s. 19493255). IEEE. doi:https://doi.org/10.1109/APSAR46974.2019.9048519

Laban, N., Abdellatif, B., Ebeid,, H. M., Shedeed, H. A., & Tolba, M. F. (2020). Convolutional Neural Network with Dilated Anchors for Object Detection in Very High Resolution Satellite Images. *2019 14th International Conference on Computer Engineering and Systems (ICCES)* (s. 19532746). IEEE. doi:https://doi.org/10.1109/ICCES48960.2019.9068145

Lee, W. Y., Parka, S. M., & Sima, K. B. (2018). Optimal hyperparameter tuning of convolutional neural networks based on the parameter-setting-free harmony search algorithm. *journal homepage, 172*, 359-367. doi:https://doi.org/10.1016/j.ijleo.2018.07.044

Maity, M., Banerjee, S., & Chaudhuri, S. S. (2021). Faster R-CNN and YOLO based Vehicle detection: A Survey. IEEE. doi:https://doi.org/10.1109/ICCMC51019.2021.9418274

MAO, Q. C., SUN, H. M., LIU, Y. B., & JIA, R. S. (2019). Mini-YOLOv3: Real-Time Object Detector for Embedded Applications. 133529 - 133538. doi:https://doi.org/10.1109/ACCESS.2019.2941547

Nie, G. H., Zhang, P., Niu, X., Dou, Y., & Xia, F. (2017). Ship detection using transfer learned single shot multi box detector. *ITM Web of Conferences, 12,2017,* s. 6. doi:https://doi.org/10.1051/itmconf/20171201006

Pandey, D., Niwaria, K., & Chourasia, B. (2020). Machine Learning Algorithms - A Review. *International Journal of Science and Research (IJSR), 9*. doi:https://www.academia.edu/download/59817925/IRJET-V6I217620190621-73452-h5pwu7.pdf

Pritt, M., & Chern, G. (2017). Satellite Image Classification with Deep Learning. (s. 18078958). IEEE. doi:https://doi.org/10.1109/AIPR.2017.8457969

Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. doi:https://doi.org/10.48550/arXiv.1804.02767

Ren, S., He, K., Girshick, R., & Sun, J. (2019). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. doi:https://proceedings.neurips.cc/paper_files/paper/2015/hash/14bfa6bb14875e45bba028a21ed38046-Abstract.html

Zhang, S., Wu, R., Xu, K., Wang, J., & Sun, W. (2019). R-CNN-Based Ship Detection from High Resolution Remote Sensing Imagery. *remote sensing*. doi:https://doi.org/10.3390/rs11060631

Alganci, U., Soydas, M., & Sertel, E. (2020, february 1). Comparative Research on Deep Learning Approaches for Airplane Detection from Very High-Resolution

Satellite Images. *remote sensing, 12*, s. 12, 458. doi:https://www.mdpi.com/2072-4292/12/3/458

Bahi, M., & Batouche, M. (2018). Deep Learning for Ligand-Based Virtual Screening in Drug Discovery. IEEE. doi:https://www.researchgate.net/publication/330120030_Deep_Learning_for_Ligand-Based_Virtual_Screening_in_Drug_Discovery

Claesen, M., & Moor, B. D. (2015). Hyperparameter Search in Machine Learning., (s. 2446, 3001). doi:https://doi.org/10.48550/arXiv.1502.02127

Cordova, A. W., Quispe, W. C., Inca, R. J., Choquehuayta, W. N., & Gutierrez, E. C. (2020). New Approaches and Tools for Ship Detection in Optical Satellite Imagery. *Journal of Physics: Conference Series*, 1642. doi:https://iopscience.iop.org/article/10.1088/1742-6596/1642/1/012003

Cordova, A. W., Quispe, W. C., Inca, R. J., Choquehuayta, W. N., & Gutierrez, E. C. (2020). New Approaches and Tools for Ship Detection in Optical Satellite Imagery. doi:https://iopscience.iop.org/article/10.1088/1742-6596/1642/1/012003/meta

Li, J., Gu, J., Huang, Z., & Wen, J. (2019). Application Research of Improved YOLO V3 Algorithm in PCB Electronic Component Detection. doi:https://doi.org/10.3390/app9183750

Li, X., Wang, S., Jiang, B., & Chan, X. (2022). Airplane Detection Using Convolutional Neural Networks in a Coarse-to-fine Manner. (s. 17576497). IEEE xplore. doi:https://doi.org/10.1109/ITNEC.2017.8284943

Li, Y., Fu, K., Sun, H., & Sun, X. (2018). An Aircraft Detection Framework Based on Reinforcement Learning and Convolutional Neural Networks in Remote Sensing Images. *remote sensing*. doi:https://www.mdpi.com/2072-4292/10/2/243

Lin, H., Shi, Z., & Zou, Z. (2017). Fully Convolutional Network With Task Partitioning for Inshore Ship Detection in Optical Remote Sensing Images. *14*, 17205845. doi:https://doi.org/10.1109/LGRS.2017.2727515

Liu, Y., Liu, N., Huo, H., & Fang, T. (2022). Vehicle Detection in High Resolution Satellite Images with Joint-Layer Deep Convolutional Neural Networks. (s. 16618793). IEEE. doi:https://doi.org/10.1109/M2VIP.2016.7827266

Saravanan, R., & Sujatha, P. (2019). A State of Art Techniques on Machine Learning Algorithms: A Perspective of Supervised Learning Approaches in Data Classification. IEEE. doi:https://doi.org/10.1109/ICCONS.2018.8663155

Thoudoju, A. K., & Kusetoğulları, D. H. (2021). Detection of Aircraft, Vehicles and Ships in Aerial and Satellite Imagery using Evolutionary Deep Learning. doi:https://www.diva-portal.org/smash/record.jsf?pid=diva2:1609134

RESUME

Personal Information

Surname, name	: Ayoob, Othman Mohammed Ayoob
Nationality	: Iraqi

Education

Degree	Education Unit	Graduation Date
Master	Electrical and Electronics Engineering	19/07/2023
Bachelor	Electrical Engineering	07/01/2019
High School	Scientific Section	2014/2013

Work Experience

Year	Place	Title
10	Iraq Anbar	Typical roads CO. for general trading contracting & transportation

Foreing Language

English & Arabic

Publications

Hobbies