Identifying ships Using Satellite Images Using Deep Learning Techniques

A Project Report submitted in the partial fulfillment of the

Requirements for the award of the degree

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by

T. Rukmini Gayathri (20471A0561)

M. Asritha (20471A0535)

B. Sai harsha (20471A0506)

Under the esteemed guidance of

Dr. Sireesha.M,B.Tech, M.Tech., Ph.D

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NARASARAOPETA ENGINEERING COLLEGE:NARASARAOPET

(AUTONOMOUS)

Accredited by NAAC with A+ Grade and NBA under Tyre -1 NIRF
rank in the band of 251-320 and an ISO 9001:2015 Certified

Approved by AICTE, New Delhi, Permanently Affiliated to JNTUK, Kakinada

KOTAPPAKONDA ROAD, YALAMANDA VILLAGE, NARASAROPET- 522601

2023 - 2024

NARASARAOPETA ENGINEERING COLLEGE:NARASARAOPET

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project work entitled "Identifying Ships Using Satellite Images Using Deep Learning Techniques" is a bonafide work done by the team T.Rukmini Gayathri(20471A0561), M.Asritha(20471A0535), B.Sai Harsha(20471A0506)in partial fulfillment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in the Department of COMPUTER SCIENCE AND ENGINEERING during 2023-2024.

PROJECT GUIDE	PROJECT CO-ORDINATOR
Sireesha.M,B.Tech, M.Tech., Ph.D.,	**Sireesha.M,B.Tech, M.Tech.Ph.D.,**
Assoc.Professor	**Assoc. Professor**

HEAD OF THE DEPARTMENT	EXTERNAL EXAMINER
Dr. Tirumala Rao.S.N,B.Tech,M.Tech,Ph.D.,	
Professor & HoD	

DECLARATION

We declare that this project work titled “IDENTYFING SHIPS USING SATELLITE IMAGES USING DEEP LEARNING TECHNIQUES” is composed by ourselves that the work contain here is our own except where explicitly stated otherwise in the text and that this work has been submitted for any degree or professional qualification except as specified.

T.Rukmini Gayathri(20471A0561)
M.Asritha(20471A0535)
B.Sai Harsha(20471A0506)

ACKNOWLEDGEMENT

We wish to express our thanks to carious personalities who are responsible for the completion of the project. We are extremely thankful to our beloved chairman sri **M.V.Koteswara Rao**,**B.Sc.**, who took keen interest in us in every effort throughout this course. We owe out sincere gratitude to our beloved principal **Dr.Sreenivasa Kumar.M, M.Tech., Ph.D., MISTE., FIE(I).,** for showing his kind attention and valuable guidance throughout the course.

We express our deep felt gratitude towards **Dr.Tirumala Rao.S.N, M.Tech.,Ph.D.,** Professor & Head of CSE department and also to our guide **Dr.Sireesha.M,B.Tech,M.Tech.,Ph.D.,**Associative Professor of CSE department whose valuableguidance and unstinting encouragement enable us to accomplish our project successfully in time.

We extend our sincere thanks towards **Ms Sireesha.M,B.Tech,M.Tech.,Ph.D.,** Associate professor & Project coordinator of the project for extending her encouragement. Their profound knowledge and willingness have been a constant source of inspiration for us throughout this project work.

We extend our sincere thanks to all other teaching and non-teaching staff department for their cooperation and encouragement during our B.Tech degree.

We have no words to acknowledge the warm affection, constant inspiration and encouragement that we received from our parents.

We affectionately acknowledge the encouragement received from our friends and those who involved in giving valuable suggestions had clarifying out doubts which had really helped us in successfully completing our project.

By

T. Rukmini Gayathri (20471A0561)
M. Asritha (20471A0535)
B. Sai Harsha (20471A0506)

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

INSTITUTE VISION AND MISSION

INSTITUTION VISION

To emerge as a Centre of excellence in technical education with a blend of effective student centric teaching learning practices as well as research for the transformation of lives and community,

INSTITUTION MISSION

M1: Provide the best class infra-structure to explore the field of engineering and research

M2: Build a passionate and a determined team of faculty with student centric teaching, imbibing experiential, innovative skills

M3: Imbibe lifelong learning skills, entrepreneurial skills and ethical values in students for addressing societal problems

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION OF THE DEPARTMENT

To become a centre of excellence in nurturing the quality Computer Science & Engineering professionals embedded with software knowledge, aptitude for research and ethical values to cater to the needs of industry and society.

MISSION OF THE DEPARTMENT

The department of Computer Science and Engineering is committed to

M1: Mould the students to become Software Professionals, Researchers and Entrepreneurs by providing advanced laboratories.

M2: Impart high quality professional training to get expertize in modern software tools and technologies to cater to the real time requirements of the Industry.

M3: Inculcate team work and lifelong learning among students with a sense of societal and ethical responsibilities.

Program Specific Outcomes (PSO's)

PSO1: Apply mathematical and scientific skills in numerous areas of Computer Science and Engineering to design and develop software-based systems.

PSO2: Acquaint module knowledge on emerging trends of the modern era in Computer Science and Engineering

PSO3: Promote novel applications that meet the needs of entrepreneur, environmental and social issues.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Educational Objectives (PEO's)

The graduates of the programme are able to:

PEO1: Apply the knowledge of Mathematics, Science and Engineering fundamentals to identify and solve Computer Science and Engineering problems.

PEO2: Use various software tools and technologies to solve problems related to academia, industry and society.

PEO3: Work with ethical and moral values in the multi-disciplinary teams and can communicate effectively among team members with continuous learning.

PEO4: Pursue higher studies and develop their career in software industry.

Program Outcomes

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

NARASARAOPETA
NEC ENGINEERING COLLEGE
(AUTONOMOUS)

Project Course Outcomes (CO'S):

CO425.1: Analyse the System of Examinations and identify the problem.

CO425.2: Identify and classify the requirements.

CO425.3: Review the Related Literature

CO425.4: Design and Modularize the project

CO425.5: Construct, Integrate, Test and Implement the Project.

CO425.6: Prepare the project Documentation and present the Report using appropriate method.

Course Outcomes – Program Outcomes mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C421.1		√											√		
C421.2	√		√		√								√		
C421.3				√		√	√	√					√		
C421.4			√			√	√	√					√	√	
C421.5					√	√	√	√	√	√	√	√	√	√	√
C421.6									√	√	√		√	√	

Course Outcomes – Program Outcome correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C421.1	2	3											2		
C421.2			2		3								2		
C421.3				2		2	3	3					2		
C421.4			2			1	1	2					3	2	
C421.5					3	3	3	2	3	2	2	1	3	2	1
C421.6									3	2	1		2	3	

Note: The values in the above table represent the level of correlation between CO's and PO's:

1. **Low level**

2. **Medium level**

3. **High level**

Project mapping with various courses of Curriculum with Attained PO’s:

Name of the course from which principles are applied in this project	Description of the device	Attained PO
C2204.2,C22L3.2	Gathering the requirements and defining the problem, plan to develop a model for identifying ships using satellite images using yolov3 algorithm	PO1, PO3
CC421.1,C2204.3,C22L3.2	Each and every requirement is critically analyzed, the process model is identified and divided into seven modules	PO2, PO3
CC421.2,C2204.2,C22L3.3	Logical design is done by using the unified modelling language which involves individual team work	PO3, PO5, PO9
CC421.3,C2204.3,C22L3.2	Each and every module is tested, integrated, and evaluated in our project	PO1, PO5
CC421.4,C2204.4,C22L3.2	Documentation is done by all our four members in the form of a group	PO10
CC421.5,C2204.2,C22L3.3	Each and every phase of the work in group is presented periodically	PO10, PO11
C2202.2,C2203.3,C1206.3, C3204.3,C4110.2	Implementation is done and the project will be handled by the satellite image analysis team and further updates in our project can be done using video inputs as well	PO4, PO7
C32SC4.3	The physical design includes website to check whether ship is present within an image or not	PO5, PO6

ABSTRACT

When using remote sensing pictures formarine security, ship detection is essential. The deep learning method for identifying ships from satellite photos is covered in this research. In order to achieve integrity Hashing is included. This model makes use of asupervised method for classifying images, and then use You Only Look Once version 3 (YOLOv3) for object recognition and feature extraction from deep CNN. Using class labels, semantic segmentation and picture segmentation are used to determine the object category of each pixel. Next, with the satellite image's bounding box is defined and helps us to identify the position of shipand ship count. Then, the concept of hashing using SHA-256 is applied in conjunction with the ship count and location of bounding box in satellite image.A dataset of around 2,30,000 photos from Kaggle Ships is used to test the suggested model.

30% of the data is used for testing, while the remaining 70% is used for training. The bounding box location and the ship count are the input data used by the hash algorithm. The proposed model achieves integrity byusing SHA-256. This model allows secure transmission highly confidential images. In our investigation ,we have achieved the accuracy score of 97% using yolov3 algorithm.

INDEX

1. INTRODUCTION

"The detection of ships using satellite images" refers to the process of identifying and locating ships within images captured by satellites orbiting the Earth. This capability is integral to various applications, including maritime surveillance, naval operations, maritime traffic monitoring.

Humans have an extraordinary ability to effortlessly spot and recognize objects in their surroundings, a feat not easily replicated by machines. Unlike humans, computers require substantial computational power and energy to perform similar tasks[1]. When tasked with identifying objects inimages or videos, machines go through a series of steps: they first extract features from the object, then process these features, and finally, determine the object's identity.

Object detection holds immense value across various domains, including online image search, object tracking, vehicle navigation, and machine-driven investigations. However, achieving accurateobject detection poses challenges for machines, such as variations in lighting conditions, positioningthe objects,and their sizes, which can significantly impact accuracy.

Satellite imagery presents a rich source of information, comprised of numerous pixels that depict land, weather patterns, and even maritime activities. Detecting ships within satellite images is particularly challenging yet critical for national security and maritime surveillance. This process involves analyzing extensive datasets to pinpoint the presence and location of ships amidst diverse environmental conditions.

Satellite images show us what's happening on Earth, including ships at sea. Finding these ships is important for keeping an eye on our waters and staying safe. For example, it is still difficult to guarantee the reliability and robustness of the object model obtained from an image due to the interference caused by different perspectives and scenarios from where the image was captured [2].

These programs analyze patterns and anomalies in the images, using algorithms to detect shapes and movements that might indicate the presence of ships. By combining advanced technology with satellite imagery, we can efficiently monitor maritime activities and ensure maritime security

Lots of smart ideas have been suggested for finding objects in images, but they often struggle with being super accurate and fast at the same time. That's why a new way to spot ships using satellite images is being talked about. This new method is extra good because:

1. It uses a system called YOLOv3 that's great at finding objects of different sizes.
2. It adds extra security to the process by using something called hashing, which makes it hard for bad guys to mess with the data when it's being sent around.

Yolov3 algorithm improves localization accuracy and multi-scale feature extraction to detect objects at three different levels. However, the proposed algorithm is a better trade-off between real-time performance and detection accuracy, and is more suitable for actual scenes[3]

With efficient training and inference, YOLOv3 achieves high-speed object detection while maintaining competitive accuracy levels.So before going deep into the model building,we need to understand the techniques used in building the model.

Deep learning is super important because it helps computers understand and learn from lots of information, just like humans do. Specifically, the detection of oriented objects from satellite photography is a complex task since the targeted objects can be visible from arbitrary directions and they are normally tightly packed. Recently, arbitrary-oriented object detection has received a lot of attention since they often appear in natural scenes, satellite photos[4].Deep learning, characterized by neural networks with multiple layers, is instrumental in various fields dueto its ability to analyze vast and complex datasets.As technology continues to evolve, deep learning remains at the forefront, driving transformative change and shaping the future landscape of artificial intelligence and beyond.In object detection, deep learning techniques such as convolutional neural networks (CNNs) have significantly improved accuracy and efficiency by automatically learning hierarchical representations of data.

Specifically in ship detection from satellite images, deep learning models like YOLOv3 excel in identifying ships amidst diverse environmental conditions, contributing to enhanced maritime surveillance, security, and environmental monitoring efforts.

We used YOLOv3 algorithm to train the model.YOLOv3 (You Only Look Once version 3) is a supervised learning algorithm. In supervised learning, the algorithm is trained on a labeled dataset, where both input data and corresponding output labels are provided during training.

Various techniques and methods are included in our model to detect ships with high accuracy.The data preprocessing techniques like Normalization,Percentile removal and Downsampling techniques are used. Data preprocessing techniques involve preparing the data for analysis or modeling by cleaning, transforming, or scaling it. They focus on enhancing the quality and usability of the data forsubsequent tasks.

Data manipulation techniques like Group By Function,Reset Index,rel2bbox function,drop function and various hashing techniques are implemented in our model. Data manipulation techniques involve restructuring or modifying the data to better suit the requirements of the analysis or modeling process. They involve operations such as grouping, indexing, encoding, dropping, or hashing to organize, represent, or process the data effectively.

As a supervised learning algorithm trained on labeled datasets, YOLOv3 forms the cornerstone ofour approach. Additionally, the model incorporates a suite of techniques encompassing both data preprocessing and manipulation. Data preprocessing endeavors to refine and enhance the quality andusability of the data through tasks such as normalization and percentile removal. Upon detecting shipswithin satellite imagery, the next step involves enumerating the total number of vessels present in a given area. This process draws upon concepts from mathematical analysis and algorithmic design, where algorithms are developed to systematically count and tally individual ships. Utilizing image segmentation and object tracking techniques, our model identifies and delineates each ship, incrementing a counter for every vessel detected. Through this approach, we obtain precise counts ofships within specified regions of interest, providing valuable insights into maritime traffic density anddistribution.

Furthermore, to enhance the security and integrity of maritime surveillance data, each detected ship is assigned a unique hash value[5]. Hash functions, rooted in cryptographic principles, transform ship attributes and characteristics into fixed-length hash codes. These hash values serve as digital signatures, uniquely identifying each ship based on its visual properties. By employing secure hashing algorithms, such as SHA-256, we ensure that each hash value is resistant to tampering and manipulation, thereby safeguarding the authenticity and reliability of ship identification data.

In conclusion, ship detection through satellite imagery, ship enumeration, and hash value generation constitute integral components of our maritime surveillance system. Drawing upon principles from computer vision, mathematics, and cryptography, our approach enables accurate andreliable identification,and tracking of ships in maritime environments. By harnessing advanced technologies and theoretical frameworks, we aim to bolster maritime security, enhanceenvironmental monitoring, and safeguard the integrity of maritime data on a global scale.

The following model detects the prediction area from the satellite images with a higher accuracy compared to other existing ones. It also works best under images with adverse climatic conditions[6]. Furthermore, ongoing research and development efforts are focused on continuously refining the model's performance and expanding its applicability to diverse maritime scenarios, ensuring its relevance and effectiveness in addressing evolving challenges in maritime domain awareness.

Looking ahead, our vision extends beyond mere detection to encompass comprehensive situational awareness in maritime environments. By integrating advanced sensor technologies, machine learning algorithms, and real-time data analytics, we aim to create a robust maritime intelligence platform capable of providing actionable insights for decision-makers. This platform will not only facilitate the early detection of potential threats but also support proactive measures to mitigate risks and enhance maritime safety. Through continued innovation and collaboration, we are committed to shaping the future of maritime security and surveillance, ensuring a safer and more secure maritime domain.

Continuous monitoring and adaptation are central to our strategy. We remain vigilant to emerging trends and evolving threats in the maritime domain, proactively refining our model to stay ahead of new challenges. By integrating sustainable practices into our development process, we aim to contribute to the preservation of marine ecosystems while enhancing maritime security and surveillance capabilities. We prioritize transparency and accountability in our methodologies, ensuring that the benefits of our technology are balanced with the protection of privacy and adherence to ethical standards[7]

1.1 Existing System

While traditional object detection systems have been widely used and can perform well under certain conditions, they often struggle with complex and diverse datasets due to their reliance on manually crafted features[8]. Moreover, the effectiveness of these systems heavily depends on the quality of the features selected and the robustness of the classifiers employed. As a result, they may encounter limitations when faced with variations in object appearance, scale, and occlusion. In contrast, deep learning-based approaches, such as convolutional neural networks (CNNs), have demonstrated remarkable success in object detection tasks by automatically learning discriminative features from raw data. CNN-based classification models have almost dominated the field of deep learning image classification, and their accuracy rates surpass those of traditional methods[9]

Additionally, deep learning models can leverage large-scale datasets to learn intricate patterns and relationships, further enhancing their capability to detect objects accurately across a wide range of conditions[10]. Despite their reliance on manually crafted features, these methods have been widely used and have shown reasonable performance in various object detection tasks. However, they may struggle with complex scenarios or require extensive tuning to achieve optimal results. However,these traditional methods have several disadvantages:

1. Limited adaptability to diverse environmental conditions, lighting changes, and object orientations reduces accuracy in complex scenarios.
2. Manual feature selection for object detection is time-consuming and may overlook relevant information in diverse datasets.
3. Poor generalization with new datasets or object types leads to subpar performance.
4. Scalability challenges arise when processing large data volumes or performing real-time detection tasks.
5. High maintenance costs and time-consuming updates for keeping systems current.
6. Difficulty accurately detecting objects in cluttered screens, resulting in higher error rates.

1.2 Proposed System

Our system relies on YOLOv3, a powerful tool for detecting ships in satellite images. YOLOv3 works like a smart detective that can quickly spot ships in pictures. It's fast, accurate, and can handle different types of ships and tricky situations like bad weather or weird lighting. With YOLOv3, we can efficiently find ships in satellite images.The several advantages of our proposed system are:

1. YOLOv3 enables real-time object detection with rapid processing of images.
2. It maintains high precision in object detection tasks, even in complex scenes.
3. YOLOv3 detects objects of different sizes within the same image.
4. It can be customized for various datasets and detection requirements.
5. YOLOv3 simplifies the detection pipeline and reduces computational overhead.
6. The model supports simultaneous training of the entire network.
7. YOLOv3 seamlessly integrates into existing systems and frameworks.

There are seven modules in this project where each module performs certain task.In the first module,the data is obtained from Kaggle airbus ship detection dataset.

The second module focuses on identifying unique images to enhance efficiency and accuracy. This isachieved using the group by function, followed by resetting the index of the entire dataframe. The 'inplace' attribute, if set to True, ensures that changes are made directly to the original dataframe.

The third module involves declaring a function for encoding bounding boxes. The 'rel2bbox' function is created to detect bounding boxes using encoded pixels. It ensures accurate bounding boxes. detection for any detected ships, raising a value error if there's any deviation in image dimensions. The function returns four values, effectively tying ships and bounding boxes together.

The Fourth module deals with normalizing and dropping encoded pixels. After defining the bounding box function, normalization of encoded pixels is necessary, followed by finding bounding boxes for these normalized-encoded pixels. Encoded pixels without corresponding bounding boxes are dropped from the dataset using the 'drop' function from the Pandas library[11].

The fifth module removes boxes with areas less than the 1st percentile, as well as those with zero area. This preprocessing step addresses the high data imbalance in the dataset, which is visualized toshow that most ships are small, and many images contain only one ship.

In Sixth module, downsampling is performed to handle the large volume of data. Only 1000 images from each class are considered for evaluation, reducing the dataset size. The dataframe index is reset, leaving approximately 8000 images for training.

The seventh module introduces hashing based on ship count and bounding box position within the image dimensions. This hashing mechanism generates a hash value displayed atop the output, enhancing security during image transmission.

2.LITERATURE SURVEY

2.1DeepLearning

Deep learning is a groundbreaking approach to artificial intelligence that has revolutionized various fields by enabling computers to automatically learn from data. Unlike traditional methods that require handcrafted features, deep learning algorithms learn hierarchical representations directly from raw data, often surpassing human-level performance in tasks like image recognition, speech processing, and natural language understanding. This is made possible by deep neural networks, which consist of interconnected layers of artificial neurons inspired by the structure of the human brain.

The input layer processes and passes the data to layers further in the neural network. These hidden layers process information at different levels, adapting their behavior as they receive new information. Deep learning networks have hundreds of hidden layers that they can use to analyze a problem from several different angles. The output layer consists of the nodes that output the data. Deep learning models that output "yes" or "no" answers have only two nodes in the output layer.

However, deep learning also presents challenges such as the need for large labeled datasets, significant computational resources, and potential interpretability issues with complex models. Despite these challenges, ongoing research efforts continue to refine deep learning algorithms, addressing limitations and unlocking new capabilities. As deep learning continues to evolve, its impact is expected to grow, fueling advancements in areas ranging from healthcare and autonomous vehicles to finance and beyond, ultimately reshaping the way we interact with technology and the world around us.

2.2 Some Deep Learning methods

Deep Learning algorithms are often categorized as supervised and unsupervised.

Supervised Deep Learning algorithms utilize labeled datasets, where each input is associated with a corresponding output label. These algorithms, often based on neural networkarchitectures like convolutional or recurrent networks, aim to learn mappings from inputs to outputs. They employ loss functions to measure the disparity between predictions and true labels, optimizing parameters through backpropagation and gradient descent during training. Data is typically divided into training and validation sets for model development and evaluation. After training, models are assessed on validation data for performance metrics likeaccuracy or mean squared error. Supervised deep learning finds applications in tasks such as image classification, sentiment analysis, and speech recognition, leveraging labeled data forpredictive modeling.

In contrast, **UnSupervised Deep Learning algorithms** operate on datasets lacking explicit labels, relying instead on the intrinsic structure or patterns within the data. Common unsupervised algorithms include autoencoders, which aim to reconstruct input data, and generative adversarial networks (GANs), which generate new samples resembling the training data distribution. These algorithms learn to capture the underlying structure of the data withoutexplicit supervision. They find applications in tasks like dimensionality reduction, data denoising, clustering, and generating synthetic data. Unsupervised learning is particularly useful when labeled data is scarce or expensive to obtain, enabling the discoveryof meaningfulrepresentations and insights from unannotated datasets.

Semi-supervised Deep Learning algorithms leverage both labeled and unlabeled data during training to improve model performance. These algorithms typically begin by training on the small labeled dataset using supervised learning techniques. Then, they usethe larger pool of unlabeled data to further refine the model's representations through techniques such as self-training, co-training, or using semi-supervised variants of neural networks like semi-supervised GANs. By exploiting the additional unlabeled data, semi- supervised learning can help mitigate the limitations of limited labeled data and enhance the model's generalization ability. This approach is particularly useful in scenarios where acquiring labeled data is expensive or time-consuming, making semi-supervised learninga valuable tool for various real-world applications.

Reinforcement learning (RL) with deep learning involves training agents to make sequential decisions in environments, aiming to maximize cumulative rewards. Agents interact with environments, receiving feedback through rewards based on actions taken. Deep RL utilizes neural networks to approximate value functions or policies, enabling complex decision-making. Techniques like Q-learning, policy gradients, and deep Q-networks (DQN) are common. Agents learn optimal strategies through trial and error, applicable in game playing, robotics, and autonomous systems. RL excels where explicit supervision is challenging, enabling agents to learn through environment interaction.

2.3 Applications of Deep Learning

1. Image Recognition and Classification
2. Natural Language Processing (NLP)
3. Speech Recognition and Synthesis
4. Autonomous Vehicles
5. Medical Image Analysis

6. Predictive Analytics
7. Fraud Detection
8. Recommendation Systems
9. Sentiment Analysis
10. Generative Models for Creative Applications

2.4 Prevalence of ship detection

The prevalence of ship detection using satellite images has significantly increased in recent yearsdue to advancements in both satellite technology and machine learning algorithms. With the proliferation of high-resolution satellite imagery and the availability of powerful computational resources, researchers and organizations have been able to develop more accurate and efficient ship detection systems.

Satellite-based ship detection has found applications in various fields, including maritime security, environmental monitoring, and maritime traffic management. These systems leverage the distinctivefeatures of ships, such as their size, shape, and motion patterns, to distinguish them from the surrounding sea surface or coastline in satellite images. Additionally, machine learning techniques, particularly deep learning algorithms, have greatly improved the accuracy and speed of ship detectionby automatically learning and extracting relevant features from satellite imagery data.

Furthermore, the increasing accessibility of satellite data through platforms like Google Earth Engine and the European Space Agency's Sentinel Hub has democratized ship detection capabilities, allowing researchers, government agencies to easily access and analyze satellite imagery for ship monitoring. As a result, ship detection using satellite images has become a prevalent tool for various applications, offering insights into maritime activities and informed decision-making-process.

2.5 Importance of deep learning in remote sensing and maritime surveillance

Deep learning is really important for remote sensing and keeping an eye on the ocean. It helps us understand what's happening in places we can't easily reach, like forests or far-off oceans. In remote sensing, deep learning uses powerful computers to understand images taken from satellites or planes. It can recognize things like forests, cities, or even individual trees[5]. In maritime surveillance, deeplearning helps spot ships and boats in the vast ocean from satellite images. It's like having a big eye in the sky that can keep watch over the water, helping us track ships, detect illegal activities, and protect the environment.

Deep learning makes it easier for us to understand and monitor our planet, even in places that are hard to reach. Moreover, deep learning techniques can be combined with other remote sensing technologies such as synthetic aperture radar (SAR) and automatic identification systems (AIS) to enhance maritime surveillance capabilities further. SAR can penetrate clouds and darkness, allowing for continuous monitoring regardless of weather conditions, while AIS provides real-time informationon vessel locations and identities. By integrating these data sources with deep learning algorithms, researchers and authorities can gain a comprehensive understanding of maritime activities and respond proactively to potential threats or incidents.

2.6 Implementation of Deep Learning using Python

Python is a popular programming language. It was created in 1991 by Guido van Rossum.It is used for:

1. web development (server-side),
2. software development,
3. mathematics,
4. system scripting.

The most recent major version of Python is Python 3. However, Python 2, although not being updated with anything other than security updates, is still quite popular.

It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse, Anaconda which are particularly useful when managing larger collections of Python files.

Random: The random module in Python provides functions for generating random numbers and selections. It can be used for simulations, random sampling, shuffling, and cryptography. The functions in the random module produce pseudo-random numbers based on algorithms with finite randomness. Moreover, the random module's capabilities extend beyond simple random number generation, offering functions for generating random selections from sequences and for controlling the seed for reproducible results. By providing a reliable framework for randomness in Python, the random module facilitates the development of diverse applications across numerous domains.

Hashlib: The hashlib module provides interfaces to secure hash and message digest algorithms, such as SHA1, SHA256, MD5, etc. Hash functions take an input (or 'message') and return a fixed-size string of bytes. Hash functions are commonly used for data integrity verification, password storage, and cryptographic applications.

In the context of ship detection from satellite images, hashlib plays a crucial role in enhancing the security of the data processing pipeline. After employing algorithms like YOLOv3 for ship detection within satellite images, hashlib can be utilized to generate hash values for the resulting data before transmission or storage. By applying hashlib's hashing algorithms such as SHA-256 or MD5 to the detected ship data, a fixed-size digest (hash value) is generated, uniquely representing the original data. This process ensures data integrity and authenticity, as any modification to the data will result in a completely different hash value, preventing unauthorized tampering during transmission or storage.

Scikit-learn (sklearn): Scikit-learn is a machine learning library for Python built on top of NumPy, SciPy, and matplotlib. It provides simple and efficient tools for data mining and data analysis tasks, including classification, regression, clustering, dimensionality reduction, model selection, and preprocessing. Scikit-learn offers a consistent interface and supports integration with other Python libraries such as pandas and matplotlib.

Seaborn: Seaborn is a statistical data visualization library based on matplotlib. It provides a high-level interface for creating attractive and informative statistical graphics. Seaborn simplifies the process of creating complex visualizations such as heatmaps, violin plots, pair plots, and categorical plots. It also offers built-in themes and color palettes to enhance the aesthetics of plots. Moreover, Seaborn seamlessly integrates with Pandas data structures, making it convenient for data manipulation and visualization within the Python ecosystem. Its extensive documentation and active community.

OpenCV (cv2): OpenCV (Open Source Computer Vision Library) is an open-source computer vision and machine learning software library. It provides a wide range of algorithms for image processing, computer vision tasks, and machine learning. OpenCV supports various programming languages, including Python, C++, and Java, and runs on multiple platforms, including Windows, Linux, macOS,and
mobile platforms.

Matplotlib: Matplotlib is a comprehensive plotting library for Python. It provides a MATLAB-like interface for creating static, interactive, and animated visualizations. Matplotlib supports various types of plots, including line plots, scatter plots, bar plots, histograms, pie charts, and 3D plots. It offers fine-grained control over the appearance and layout of plots and can be integrated seamlessly with other libraries such as NumPy and pandas.

tqdm: tqdm is a Python library that adds a progress bar to loops and iterables. It provides a simple and customizable way to monitor the progress of lengthy operations or iterations. tqdm displays a progress bar with an estimated time remaining and can be easily integrated into scripts, notebooks, and command-line interfaces.

Shutil: The shutil module provides a higher-level interface for file operations and manipulation in Python. It offers functions for copying files, moving files, archiving, and directory operations. Shutil simplifies common file-related tasks and provides cross-platform compatibility for file operations. Additionally, the shutil module supports operations such as renaming files, setting file permissions, and deleting files or directories. Its intuitive interface and comprehensive functionality make it a valuable resource for streamlining file management tasks in Python applications.

2.7 Deep learning products

KunZhaoet al. [12] developed a model for small-object detection on remote sensing images.Two CNNmodels, namely one-stage and two-stage object detectors, were used in image classification for object detection. The main pitfall in this analysis is that one dimension clusters

were not modelled appropriately.

Ren Ying et al. [13] developed a model for detecting airplanes and ships around the world through remote sensing images. This paper discusses a model on NVIDIA TX2 and YOLOv3algorithm. The main drawback in this analysis is that the detection speed is comparatively more when compared to other popular models.

Liming Zhou et al. [14] developed a model on Multi-scale Detection Network (MSDN). It can detect small-scale aircrafts even during background noise. The main drawback in this analysis isthat the border prediction and the co-ordinates may not have high accuracy all the times.

Gang Tang et al. [15] proposed a methodology named N-YOLO, which consists of Noise Level Classifier (NLC) and SAR-target Potential Area Extraction Module (STPAE) in addition to a YOLO detection module, which helps in building a model that can perform competitively with respect to several CNN algorithms. This method has good performance for ship detection using SAR images. Its applications can be extended to territories of marine monitoring, shore to ship identification, etc. The main limitations of this model is that the ship edge information is damaged because of N-YOLO.

Lichuan Zou et al. [16] proposed a model in which multi-scale Wasserstein Auxiliary Classifier Generative Adversarial Networks (MW-ACGAN) is used along with YOLOv3. This model helps to generate an improved network that can be used to generate high resolution SAR images. The model shows an accuracy of about 94%, which is more highly effective than YOLO. The pitfalls of this model are that (i) it performs effectively only on small sample size, and (ii) it is effective only under particular sea conditions, incident angles, and polarization modes.

Nie Xinet al. [17] proposed a model in which the networkprediction layer includes the prediction box uncertain regression. Negative logarithm likelihood function and improved binary cross entropy functions are together used for redesigning the loss function. K-Means clustering technique is primarily used. The Non-Maximum Suppression (NMS) algorithm is used along with Gaussian Soft Threshold function for prediction of boxes. The only limitation is that when given large sets of input data, the model gives a varying accuracy.

Lena Chang et al. [18] proposed a method in which ships can be managed during day and night. The proposed model uses multi-scale feature extraction, which is extremely helpful in recognizing small targets. The training data set has six different categories of ships, which sums up to 5513 visible and IR images collected from various harbors in northernm Taiwan.

3. SYSTEM ANALYSIS

3.1 Scope of the project

The project's primary focus is on creating algorithms to identify ships within maritime regions using satellite imagery data. It involves acquiring and preprocessing images, implementing detectionalgorithms, and extracting ship features for analysis, such as size and speed. Additionally, the project emphasizes integration with geospatial data and the development of user-friendly interfaces for visualization and interaction, aiming to provide users with efficient tools for assessing ship presenceand characteristics.

By combining advanced algorithms with satellite imagery, the project aims to enable automated ship identification and analysis. It includes stages like image acquisition, algorithm implementation, and feature extraction for ship attributes like size and speed. Integration with geospatial data and the creation of user interfaces for visualization are also key components, ensuring the project delivers accessible tools for users to evaluate ship presence and characteristics effectively.

3.2 System Requirements

3.2.1 Hardware Requirements:

Processor : Intel® Dual Core 2.0GHz minimum

Hard Disk : 1TB minimum

RAM : 8GB or more

3.2.2 Software Requirements:

Operating System : Windows 10 Server or later

Browser : Any latest browser like Chrome

Coding Language : Python(COLAB)

3.3 Analysis

The dataset used in this project is “air bus ship detection” dataset.This dataset consists of 2,31,722 images.The model is trained with 70% of the data and tested with 30% of the data for cogency. The model consists of several steps that are to be procedurally followed to obtain the desired outcome. Satellite images offer compelling information. They usually are made up of a large number of pixels in which their size range varies from tens of centimeters to tens of meters. Satellite images can be chosen from any category, such as visible imagery, infrared imagery, and water vapor imagery based on the requirement.The dataset consists of only two attributes which are used for ship detection and visualization purpose such as:

1. imageID
2. encoded pixels

3.3.1 Dataset description

Image ID: It allows us to uniquely identify and organize each image within our dataset, making it easier to manage and reference during model training and evaluation. It facilitates the annotation process by providing a reference point for labeling the presence or absence of ships in each image. Image IDs are often used in conjunction with ground truth annotations to evaluate the performance of ship detection algorithms. By comparing the detected ships' locations and sizes with the ground truth annotations for each image, metrics such as precision, recall, and accuracy can be calculated on a per-image basis. This allows for a detailed assessment of the model's performance across different images in the dataset.

Encodedpixels: These are special codes that indicate where ships are in the pictures. They help the model learn to recognize ships by showing their locations during training. When the model examinesnew images, it checks these codes to identify ships. This method teaches the model how to find shipsaccurately in satellite pictures, improving the reliability of the detection process.

In maritime surveillance, encoded pixels enable the automated detection and segmentation of maritime objects, such as ships, boats, and offshore structures, in satellite imagery. By encoding pixel regions corresponding to different maritime entities, semantic segmentation models can effectively delineate vessel boundaries and identify their spatial extent within the marine environment.

ImageId	EncodedPixels
00003e153.jpg	
0001124c7.jpg	
000155de5.j	264661 17 265429 33 266197 33 266965 33 267733 33 268501 33 269269 33 270037 33 270805 33 271573 33 272341 33 273109 33 273877 33 274645 33 275413 33
000194a2d.j	360486 1 361252 4 362019 5 362785 8 363552 10 364321 10 365090 9 365858 10 366627 10 367396 9 368165 9 368933 10 369702 10 370471 9 371240 9 372009 9
000194a2d.j	51834 9 52602 9 53370 9 54138 9 54906 9 55674 7 56442 7 57210 7 57978 7 58746 7 59514 7 60282 7 61050 9 61818 9 62586 9 63354 9 64122 9 64890 9
000194a2d.j	198320 10 199088 10 199856 10 200624 10 201392 10 202160 10 202928 10 203696 10 204464 10 205232 10 206000 10 206768 10 207536 10 208304 10 209072 10
000194a2d.j	55683 1 56451 1 57219 1 57987 1 58755 1 59523 1 60291 1
000194a2d.j	254389 9 255157 17 255925 17 256693 17 257461 17 258229 17 258997 17 259765 17 260533 17 261301 17 262068 18 262836 17 263604 17 264372 17 265140 17
0001b1832.jpg	
00021ddc3.j	108287 1 109054 3 109821 4 110588 5 111356 5 112123 6 112890 7 113657 8 114424 9 115191 10 115958 11 116725 12 117493 12 118260 13 119027 14 119794 14
00021ddc3.j	101361 1 102128 3 102896 4 103663 6 104430 9 105198 10 105209 1 105965 14 106732 14 107500 14 108267 14 109034 14 109802 14 110569 14 111336 14 112104 14
00021ddc3.j	74441 3 75207 5 75975 5 76743 5 77511 5 78280 4 79048 5 79816 5 80584 5 81352 5 82120 5 82888 6 83657 5 84425 5 85193 5 85961 2
00021ddc3.j	74444 4 75212 4 75980 4 76748 4 77517 3 78285 3 79053 4 79821 4 80589 4 81357 4 82125 4
00021ddc3.j	150423 2 151190 3 151958 3 152726 4 153495 3 154263 3
00021ddc3.j	139644 2 140408 6 141174 9 141942 9 142711 6 143479 2
00021ddc3.j	75972 3 76738 5 77506 5 78274 5 79042 6 79811 5 80579 5 81347 5 82115 5 82883 5 83651 6 84420 5 85188 5 85956 2
00021ddc3.j	86727 2 87493 4 88261 4 89030 3 89798 4 90566 4 91334 4 92103 3 92871 1
00021ddc3.j	95225 2 95992 5 96760 7 97527 9 98294 9 99062 9 99829 9 100596 9 101364 9 102132 8 102901 6 103671 3 104440 2
0002756f7.j	255784 2 256552 4 257319 7 258087 9 258854 12 259622 14 260389 16 261159 16 261929 16 262699 16 263469 16 264239 16 265009 14 265779 11 266549 9 267319 6
0002756f7.j	248878 1 249645 4 250413 6 251180 9 251948 10 252715 13 253482 16 254250 18 255019 18 255789 18 256559 18 257329 18 258099 17 258869 15 259639 12 260409
0002d0f32.jpg	
000303d4d.jpg	
00031f145.j	340363 1 341129 4 341896 6 342663 7 343429 10 344196 12 344962 15 345729 16 346496 18 347262 21 348029 23 348798 23 349567 22 350335 23 351104 23 351873
00052ed46.jpg	
000532683.j	458957 14 459725 14 460493 14 461261 14 462029 14 462797 14 463565 14
000532683.j	567543 2 568309 4 569077 5 569846 4 570614 5 571382 5 572150 6 572919 5 573687 2
00053c6ba.j	565382 3 566144 9 566909 12 567677 13 568445 13 569214 12 569982 12 570750 13 571518 13 572287 12 573055 12 573823 13 574591 13 575360 12 576128 9 576896
00057a50d.jpg	
000592296.jpg	
0005d01c8.j	56010 1 56777 3 57544 6 58312 7 59079 9 59846 11 60613 14 61380 16 62148 17 62915 19 63682 22 64449 24 65216 26 65984 27 66751 30 67518 32 68285 34 69052 36
0005d01c8.j	365871 1 366638 3 367405 6 368173 7 368940 9 369707 12 370474 14 371242 15 372009 17 372776 20 373543 22 374310 24 375078 26 375845 28 376612 30 377379 30
0005d6d95.j	265143 1 265910 4 266678 5 267445 7 268212 10 268980 11 269747 11 270514 12 271282 11 272049 11 272816 12 273583 12 274351 11 275118 12 275885 12 276653
0006c52e8.j	146366 1 147132 4 147899 5 148666 7 149432 10 150199 12 150966 13 151732 16 152499 18 153265 20 154032 22 154799 24 155565 27 156332 28 157099 30 157865
0007b8229.jpg	

Fig 3.1 Dataset

3.4 Data Pre-processing

Before feeding data to an algorithm we have to apply transformations to our data which is referred as pre-processing. By performing pre-processing the raw data which is not feasible for analysis is converted into clean data. In-order to achieve better results using a model in Deep Learning, data format has to be in a proper manner. The data should be in a particular format for different algorithms.

Pre-processing refers to the transformations applied to our data before feeding it to the algorithm. Data Pre-processing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is notfeasible for the analysis.

Preprocessing is the first step while creating the deep learning model. It is the process of converting raw dataset into cleaned dataset.It often handles complex data types like images, text, and sequences. For images, this encompasses resizing, normalization, and augmentation to enhance model robustness. Text data requires tokenization, embedding, and sequence padding for effective processing. Sequential data preprocessing, common in recurrent neural networks, involves sequence padding and handling variable-length sequences. Imbalanced data is addressed through oversampling, undersampling, or class weighting strategies. Finally, datasets are split into training, validation, and test sets, ensuring representative samples for model evaluation. Overall, data preprocessing tailored to deep learning tasks enhances model performance and facilitates accurate predictions across diverse data domains.

3.4.1 Missing values

Missing values in ship detection datasets, particularly in satellite images, pose a significant challenge during preprocessing. These missing values, often found in the Encoded Pixels column, indicate instances where no ship is detected in the corresponding image. Various factors contribute to missing values, including data acquisition issues, labeling errors, and data corruption[6]. Addressing missing values is crucial for accurate model training. Strategies such as data imputation, removal of missing instances, flagging missing values, data augmentation, and model adaptation can be employed. Each strategy has its merits and applicability depending on the dataset characteristics and analysis objectives. Handling missing values effectively ensures the reliability and robustness of ship detection models. As shown in Fig 3.1,there are few images for which there is no Encoded Pixel values which means there is no ship for those images and also there are few images for which there are multiple rows of Encoded pixel values which means there are multiple ships for these images .To handle missing values present in our dataset,the images containing no ships are removed.

Some of the deep learning techniques to handle missing values are:

1. Masking
2. Probabilistic Graphical Models (PGMs)
3. Deleting rows or columns
4. Replace with mean or median or mode value
5. Recurrent Neural networks

The Image id's which contain NaN as the encoded pixels means that there are no ships corresponding to that particular image as shown in fig 3.2.

Fig 3.3 depicts there are few images for which there is no Encoded Pixel values which means there is no ship forthose images.

ImageId	EncodedPixels
0002d0f03.jpg	NaN
0007b8229.jpg	NaN
00003e153.jpg	NaN
0001124c7.jpg	NaN
000155de5.jpg	264661 17 265429 33 266197 33 266965 33 267733...
000194a2d.jpg	360486 1 361252 4 362019 5 362785 8 363552 10 ...
000194a2d.jpg	51834 9 52602 9 53370 9 54138 9 54906 9 55674 ...
00021ddc3.jpg	150423 2 150190 3 151958 3 152726 4 153495 3…

Fig 3.2 Dataset before removing images containing no ship

ImageI	EncodedPix
000155de5.jpg	264661 17 265429 33 266197 33 266965 33 267733...
000194a2d.jpg	360486 1 361252 4 362019 5 362785 8 363552 10 ...
000194a2d.jpg	51834 9 52602 9 53370 9 54138 9 54906 9 55674 ...

Fig 3.3 Dataset after removing images containing no ship

3.5 Feature Extraction

Feature extraction in ship detection is a fundamental process where spatial attributes like ship dimensions and positions are extracted from satellite images. These details are used to precisely define bounding boxes that outline ships within the images, ensuring accuracy in identifying their locations and sizes. By addressing variations in image dimensions and potential overlap between objects, feature extraction enhances the reliability of ship detection algorithms.These encoded bounding boxes serve as vital annotations for training machine learning models, providing ground truth labels for ship locations and aiding in the extraction of essential spatial features necessary for accurate detection. In order to acquire optimal sizes of bounding boxes, the width and height of the bounding box are selected as the clustering features in K-means[19].

Moreover, feature extraction enables the extraction of contextual information surrounding the ships, such as orientations, shapes, and relative positions within the image frame. This additional context enhances the performance of ship detection algorithms by providing cues for discriminating between ships and false positives. Overall, feature extraction is a critical preprocessing step in ship detection, laying the foundation for subsequent analysis and model training, and ensuring the development of robust ship detection models capable of accurately identifying and delineating ships in satellite imagery.

3.6 Data Normalization

The normalization of encoded pixels is a pivotal step in preparing data for ship detection in satellite images. Normalization standardizes pixel values to a consistent range, aiding in stable model training. By adjusting pixel values relative to the dataset's overall distribution, normalization helps mitigate variations caused by differing lighting conditions and image resolutions. It ensures uniform feature representation, facilitating effective pattern learning by the model. Additionally, normalization promotes smoother optimization during training by preventing large parameter updates and minimizing gradient-related issues.

Furthermore, data normalization enhances the model's ability to generalize across diverse datasets by reducing the impact of outliers and scaling pixel values to a common scale. This ensures that the model learns meaningful patterns from the data, rather than being influenced by the absolute magnitude of pixel values. Overall, data normalization is an essential preprocessing step that contributes to the stability, robustness, and generalization capability of ship detection models trained on satellite imagery.

3.7 Data Analysis

Data Analysis involves examining the dataset used for ship detection from satellite images. This process includes identifying and removing small bounding boxes that may represent noise. Additionally, it visualizes ship size distribution and the number of ships per image to detect potential imbalances in the dataset. These insights inform decisions about model training strategies, such as data augmentation or class weighting, to improve model performance. Furthermore, data analysis serves as a quality assessment step, ensuring dataset representativeness and reliability for producing accurate ship detection models[20].The figure 3.4 shows the areas of Bounding boxes of ships and Bounding Box Area Distribution.

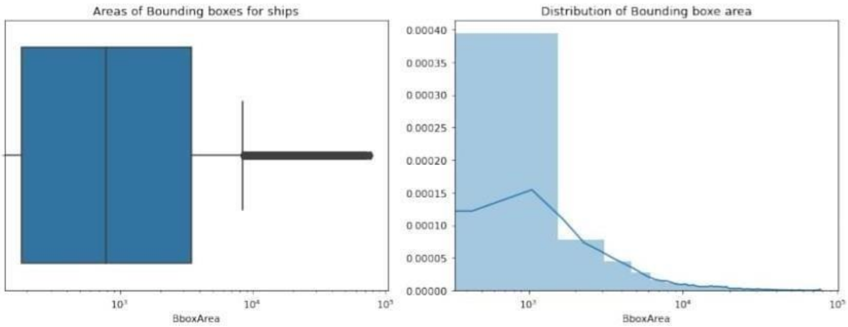

Fig 3.4 Areas of Bounding Boxes for ships

Furthermore, data normalization enhances the model's ability to generalize across diverse datasets by reducing the impact of outliers and scaling pixel values to a common scale. This ensures that the model learns meaningful patterns from the data, rather than being influenced by the absolute magnitude of pixel values. Overall, data normalization is an essential preprocessing step that contributes to the stability, robustness, and generalization capability of ship detection models trained on satellite imagery.

3.8 Downsampling:

Downsampling was used to address class imbalance within the dataset for ship detection. Initially, the dataset exhibited significant variations in sample distribution across different classes, leading to skewed representation. To mitigate this, downsampling involved selecting 1000 images from each class where the number of samples exceeded this threshold. This process ensured a more balanced representation of ship counts in the dataset. By creating a balanced dataset, the model was trained on a more equitable sample distribution, enhancing its ability to generalize and accurately detect ships in images. Downsampling played a crucial role in improving the model's performance by mitigating biases and ensuring fair representation across all classes[21].

After downsampling,there would be a reduced version of the original dataset containing fewer instances or samples ,the index of the resulting DataFrame is reset to ensure consistency and clarity in indexing. This process helps in addressing imbalanced datasets where one class dominates the others, leading to biased model performance. It also helps to avoid confusion or errors that may arise from inconsistent indexing.

In the context of computer vision and object detection, downsampling refers to the process of reducing the spatial resolution of an image. This is typically done to decrease computational complexity while preserving important features for detection tasks.

Downsampling is commonly used to create smaller versions of images for tasks like image compression and visualization. It helps in reducing the computational burden by reducing the amount of data to be processed while retaining the essential features of the image. However, downsampling may lead to loss of fine details and can affect the overall quality of the image. Various downsampling techniques exist, such as averaging, max-pooling, or nearest-neighbor interpolation, each with its advantages and drawbacks. The choice of downsampling method depends on the specific requirements of the application and the trade-offs between computational efficiency and image fidelity[22].

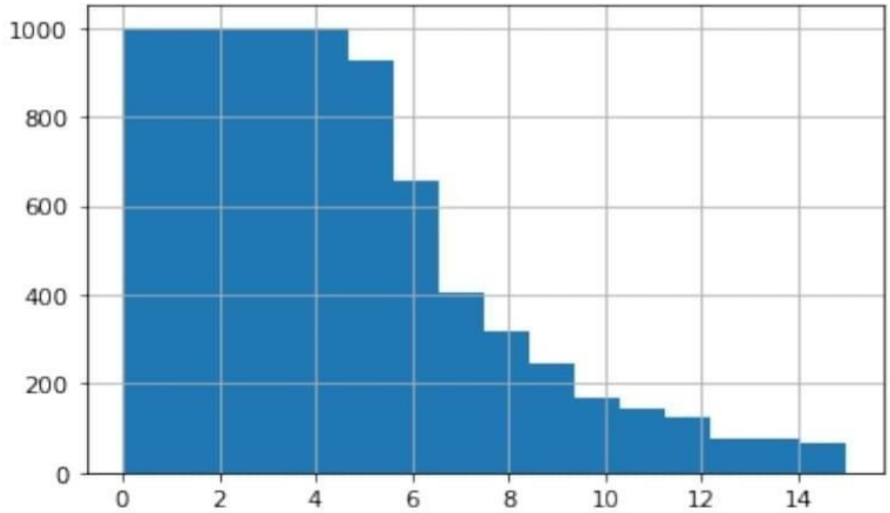

Fig 3.5 Class distribution after downsampling

3.9 Hashing

Hashing is employed based on the count of ships and the position of bounding boxes within satellite images. This hashing process generates unique hash values associated with each image, enhancing security during image transmission. By computing hash values derived from ship count and bounding box positions, the module ensures data integrity and authenticity.

Once objects are detected in the satellite images using YOLOv3, hashing techniques can be used to generate unique identifiers (hash codes) for each detected object based on its characteristics such as size, shape, and location. These hash codes can then be used for tracking, matching, or indexing purposes.

Hashing provides the ships placements and their value as a count. One of the safest and most difficult to crack algorithms, the sha-256 algorithm, is used to generate the hashing value. And it uses 512 bits and these are divided into 64 bytes and this encoded algorithm is difficult to hack and usuallytakes 2256 times for the brute force approach [24].

4.DESIGN AND ANALYSIS

The Fig 4.1 is the design of our model,firstly the dataset is taken from Kaggle,and the name of thedataset is “Air Bus Ship Detection”.

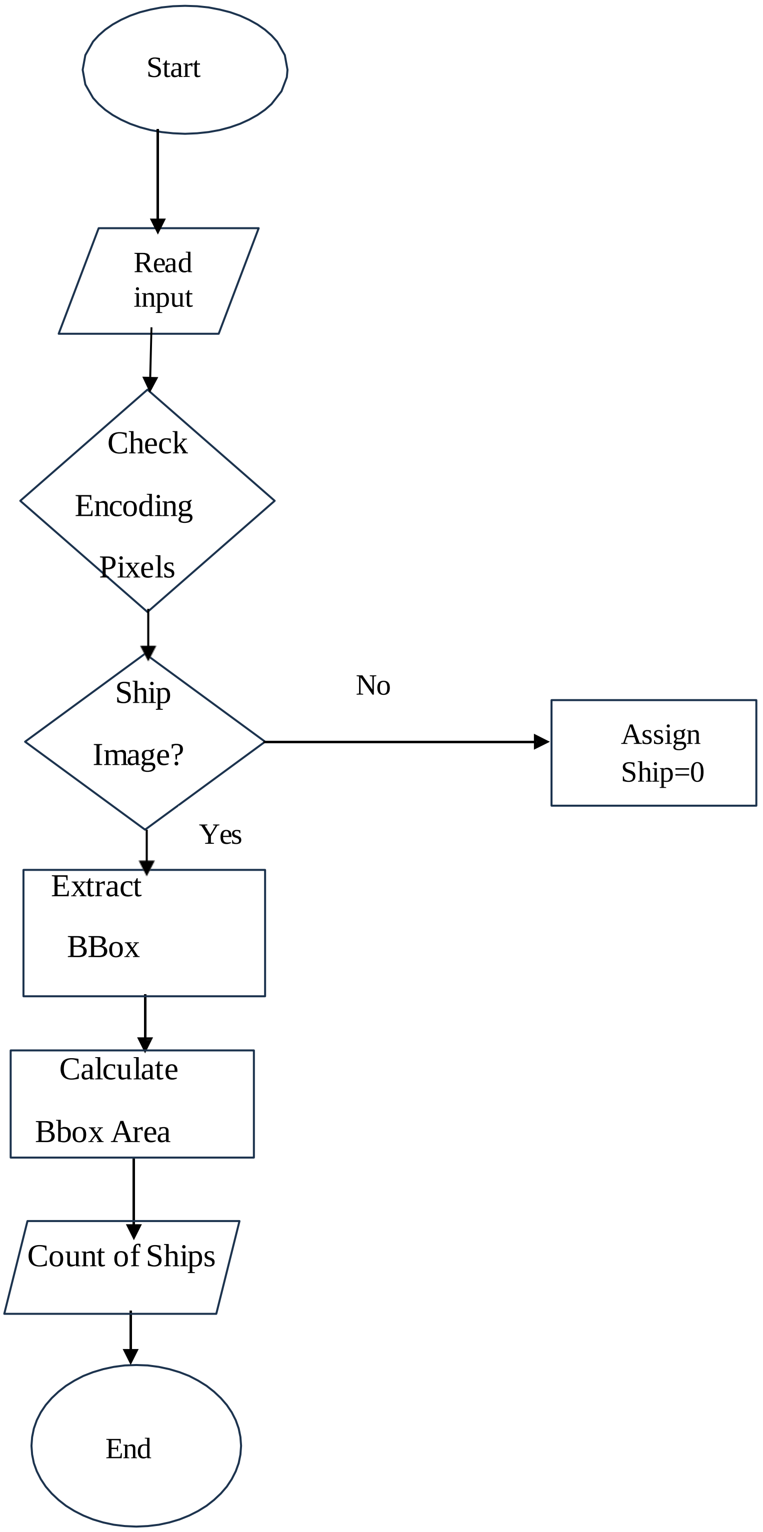

Fig 4.1 Design of our model

The Fig 4.1 outlines the key steps for analyzing and preparing the Airbus Ship Detection datasetfor training a ship detection model.

It starts by importing libraries and reading the data. Then, it checks for the presence of Encoded Pixels (indicating ships) and creates a binary classification target.

Next, the code defines a function to decode bounding boxes from the Encoded Pixels. This allows for extracting the location and size of each ship in an image. The areas of these bounding boxes are analyzed to understand the distribution of ship sizes. Finally, very small bounding boxes are removed to potentially improve model performance by eliminating noise or insignificant detections.If the ship is present,it is counted and if the ship is not present,it is assigned value zero ang then removed.

4.1 Object Detection Algorithms

Object detection algorithms are computer vision techniques designed to identify and locate multiple objects within digital images or video frames. Unlike classification algorithms, which categorize entire images into predefined classes, object detection algorithms go a step further by not only recognizing the presence of objects but also precisely outlining their boundaries with bounding boxes. These algorithms typically utilize deep learning architectures such as convolutional neural networks (CNNs) to efficiently process visual data and extract relevant features for object identification. By dividing images into grids or employing region proposal methods, object detection algorithms localize objects and assign class labels to them simultaneously, enabling applications such as autonomous vehicles, surveillance systems, and image retrieval systems to understand and interact with their visual environments effectively.

There are many object detection algorithms available but it is not possible to conclude that which is superior to other,It depends on applications and available dataset.

Some popular object detection algorithms include:

1.YOLO

2.R-CNN

3.K-Means

YOLO: The YOLO (You Only Look Once) algorithm is crucial in the field of computer vision for its ability to efficiently and accurately detect objects within images or video frames. Unlike traditional methods that require multiple passes through an image, YOLO processes the entire image in one go, making it incredibly fast.This speed makes it suitable for real-time applications like autonomous driving, surveillance, and object tracking.YOLO's approach also provides precise bounding box coordinates around detected objects and assigns them class labels simultaneously, streamlining the object detection process. Its simplicity and effectiveness have made YOLO a popular choice in various industries, driving advancements in tasks requiring fast and accurate object detection capabilities.

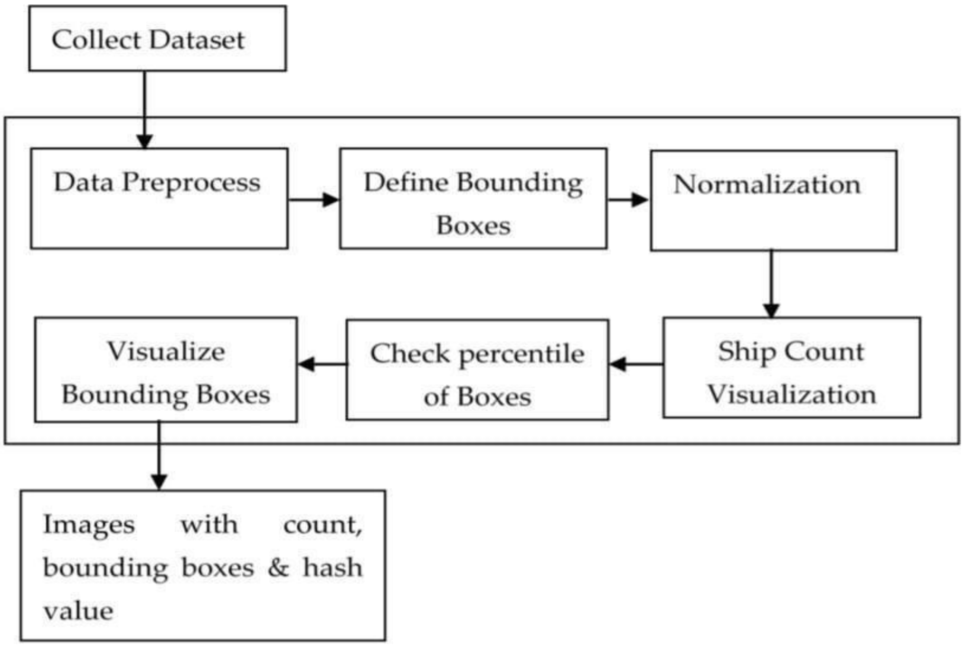

Fig 4.2 Architecture of our Proposed model

R-CNN: Region-based Convolutional Neural Networks is an important algorithm in computer visionthat revolutionized object detection tasks. Unlike earlier methods that treated object detection as a separate problem from feature extraction, R-CNN integrated both tasks into a single neural network framework. It accomplished this by proposing regions of interest within an image and then applying a convolutional neural network to extract features and classify these regions. This approach significantly improved both accuracy and efficiency in object detection tasks. R-CNN laid the foundation for subsequent advancements in the field, paving the way for faster and more accurate algorithms such as Faster R-CNN and Mask R-CNN. Its impact continues to be felt in various applications, including autonomous vehicles, surveillance systems, and image analysis tools.

In addition to the high computational cost during training, R-CNN also suffers from inefficiency during inference. Each region proposal needs to be individually processed through the CNN, leadingto redundant computations and inefficient memory usage, particularly for large images.

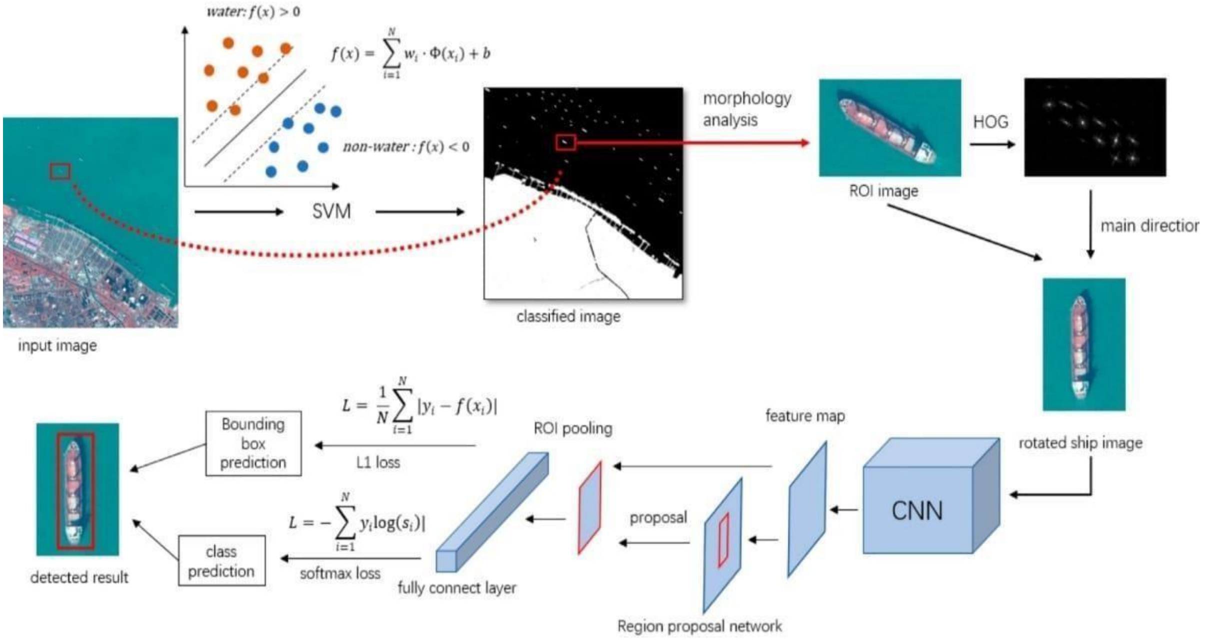

Fig 4.3 Architecture of R-CNN model

The Figure 4.3 shows the Ship detection process by using R-CNN model.

The Fig 4.3 shows if a GF2 image is given, it performs water and nonwater segmentation using SVM. In addition, obtain the ROIs that may contain ships by analyzing the size and aspect ratio of the nonwater area.It Obtains the main direction of ships in the ROI by computing the HOG feature of the ROI image. In addition, rotate the ROI image to the main direction,It also performs the R-CNN- based ship detection method on the ROI image.

K-Means: The K-means algorithm is important in object detection for its role in tasks like anchor box initialization and feature clustering. In object detection, anchor boxes are predefined bounding boxes used to detect objects of varying sizes and aspect ratios. K-means clustering helps determine the optimal anchor box sizes by partitioning the training data into clusters based on the distribution of object sizes in the dataset. These clusters then inform the selection of anchor box dimensions, ensuring that the object detector can effectively capture objects of different scales and shapes. Additionally, K-means clustering is used in feature space analysis, aiding in the extraction of representative features for object detection tasks. Overall, K-means plays a crucial role in enhancing the accuracy and efficiency of object detection algorithms by facilitating better initialization and feature representation**.** K-means helps optimize computational resources to reduce redundancy in feature representations, leading to more efficient processing. Its ability to adapt anchor box sizes based on the dataset's characteristics contributes to the robustness and generalization capabilities of object detection models.

Algorithm 1: Ship Detection

Input: Images, D.

Output: Detects the region of ships in image.

Process

Step 1: Read the dataset.

Step 2: Find unique images.

Step 3: Define function to encode bounding boxes.

Step 4: Plot the bounding box areas.

Step 5: Remove boxes that are less than 1 percentile or BoundingBoxArea < 2Pixels

Step 6: Detect the area and show the detected region of ships.

Fig 4.4 Algorithm for Ship detection

Fig 4.4 describes the process to detect ships in satellite images.Firstly we need to extract the necessary dataset required to build our model.Then further process continues to detect ships.

Algorithm2:Ship Count after Object Detection

Input: Images with the region of ship detected.

Output: Counts the number of ships in an image.

Process

Step1: Process individual images from dataset.

Step 2: Initialize count to 0.

Step 3: While the image contains more than 0 BoundingBoxes: Increment Count.

Step 4: Display count.

Fig 4.5 Algorithm of Ship Count after Object Detection

Fig 4.5 depicts the process to count how many ships are present within in a single image.It actually counts the count of bounding boxes in which ships are present.

4.2 Performance Metrics

The confusion matrix provides a comprehensive overview of the model's performance by tabulating the number of correct and incorrect predictions across different categories.

In this context, the confusion matrix helps evaluate the effectiveness of ship detection algorithms by revealing the number of true positive (correctly detected ships), true negative (correctly identified non-ships), false positive (incorrectly detected as ships), and false negative (missed ships) predictions. Analyzing the confusion matrix allows stakeholders to assess the model's accuracy, precision, recall, and other performance metrics crucial for optimizing ship detection algorithms and enhancing the reliability of satellite-based maritime surveillance systems.

Accuracy

Accuracy refers to the measure of how correctly the detection algorithm identifies and delineates ships within the images, typically expressed as the percentage of correctly identified ships compared to all ships present in the dataset.

From the confusion matrix when can calculate the accuracy it can be done by the formula of total of True Positive (TPO) and True Negative (TNN) divided by total of True Positive (TPO), True

Accuracy = (TPO+TNN)/(TPO+TNN+FPO+FNN)

The accuracy of our model by substituting the values in above formula is 97%.It means that from 100 samples, the model can absolutely predict 97 samples as correct.

Recall and Precision

Precision is about how precise or accurate your model is when it predicts positive instances. It calculates the ratio of correctly predicted positive instances (true positives) to all instances predicted as positive, including both true positives and false positives. Precision is crucial when you want to be confident that the instances your model identifies as positive are indeed positive.

Recall is defined as number of samples predicted correctly from all positive classes.
Precision is defined as the classes that are classified as positive that are actually positive.

Recall=(TPO)/(TPO+FNN)

Precision=(TPO)/(TPO+FPO)

F-Measure

F-measure in the context of ship detection using satellite images refers to a metric that combines precision and recall, providing a single measure of the model's accuracy in identifying ships while accounting for both false positives and false negatives.

F-Measure calculates both recall and precision.Let R=recall,P=precision.

F-Measure=(2*R*P)/(R+P)

5. IMPLEMENTATION

Flask code to connect Backend

```
from flask import Flask, request, send_file
import os
import cv2
import matplotlib.pyplot as plt
import hashlib
import numpy as np
import pandas as pd
from sklearn.metrics import confusion_matrix, accuracy_score, classification_report,
roc_auc_score, log_loss
import seaborn as sns
app = Flask(__name__)
ships=pd.read_csv("C:\\Users\\Tadavarthi
Gowtham\\Desktop\\Gayathri\\mainproject\\train_ship_segmentations_v2.csv")
test_data=pd.read_csv("C:\\Users\\Tadavarthi
Gowtham\\Desktop\\Gayathri\\mainproject\\sample_submission_v2.csv")

#Finding the unique images for the dataset
ships["Ship"] = ships["EncodedPixels"].map(lambda x:1 if isinstance(x,str) else 0)
ship_unique = ships[["ImageId","Ship"]].groupby("ImageId").agg({"Ship":"sum"}).reset_index()

ship_actual=0
ship_predict=0
ship_atual=ships
ship_preict=ships
X_actual = [1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 1]
for i in range(1000):
    X_actual.append(ship_actual)
for i in range(50):
    X_actual.append(1)
Y_predic = [1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 1]
for i in range(1000):
    Y_predic.append(ship_predict)
```

```
for i in range(50):
    Y_predic.append(0)

#Defining Function to encode the bounding boxes for the encoded pixels
def rle2bbox(rle, shape):
    a = np.fromiter(rle.split(), dtype=np.uint)
    # an array of (start, length) pairs
     a = a.reshape((-1, 2))
    # `start` is 1-indexed
    a[:,0] -= 1
    y0 = a[:,0] % shape[0]
    y1 = y0 + a[:,1]
    if np.any(y1 > shape[0]):
        y0 = 0
        y1 = shape[0]
    else:
        y0 = np.min(y0)
        y1 = np.max(y1)
    x0 = a[:,0] // shape[0]
    x1 = (a[:,0] + a[:,1]) // shape[0]
    x0 = np.min(x0)
    x1 = np.max(x1)
    if x1 > shape[1]:
       #just went out of the image dimensions
        raise ValueError("invalid RLE or image dimensions: x1=%d >shape[1]=%d" % (x1, shape[1]))
    xc = (x0+x1)/(2*768)
    yc = (y0+y1)/(2*768)
    w = np.abs(x1-x0)/768
    h = np.abs(y1-y0)/768
    return [xc, yc, h, w]


#Finding the bounding boxes from encoded pixels normalized
ships["Bbox"] = ships["EncodedPixels"].apply(lambda x: rle2bbox(x, (768, 768)) if isinstance(x,
str) else np.NaN)
```

```
#Dropping the Encoded pixels from the data set
ships.drop("EncodedPixels", axis=1, inplace=True)
ships["BboxArea"] = ships["Bbox"].map(lambda x: x[2]*768*x[3]*768 if x == x else 0)
area = ships[ships.Ship>0]
np.percentile(area["BboxArea"], [1, 5, 25, 50, 75, 95, 99])

#Removing boxes which are less than 1 percentile
ships = ships[ships["BboxArea"] > np.percentile(ships["BboxArea"], 1)]

#Finding the Distribution of number of ships
ship_unique["Hasship"] = [1 if x > 0 else 0 for x in ship_unique["Ship"]]

#Since the dataset volume is very high,we are downsampling to select 1000 images from each of the classes(Where more than 1000)
balanced_df = ship_unique.groupby("Ship").apply(lambda x:x.sample(1000) if len(x)>=1000 else x.sample(len(x)))
balanced_df.reset_index(drop=True,inplace=True)

#Creating data frame for bounding boxes for the images in Balanced_df
balanced_bbox = ships.merge(balanced_df[["ImageId"]], how ="inner", on = "ImageId")
balanced_bbox.head()

@app.route('/process_and_show', methods=['POST'])
def show_processed_images():
    uploaded_files = request.files.getlist('file')
    uploaded_file_paths = []
    for file in uploaded_files:
        file_path = os.path.join("uploads", file.filename)
        file.save(file_path)
        uploaded_file_paths.append(file_path)

    processed_images = process_images(uploaded_file_paths)
    return processed_images

def process_images(uploaded_files):
```

```
#Visualizing the bounding boxes and images
path = "C:\\Users\\Tadavarthi Gowtham\\Desktop\\Gayathri\\mainproject-train-data"
    plt.figure(figsize=(9, 7))

    processed_images = []

    for i, file_path in enumerate(uploaded_files):
        imageid = balanced_df[balanced_df.Ship == i].iloc[0][0]
        image_path = os.path.join(path, imageid)  # Construct the full image path

        if os.path.exists(image_path):
            image = cv2.imread(image_path)

            if i > 0:
                bbox = balanced_bbox[balanced_bbox.ImageId == imageid]["Bbox"]
                for items in bbox:
                    Xmin = int((items[0] - items[3] / 2) * 768)
                    Ymin = int((items[1] - items[2] / 2) * 768)
                    Xmax = int((items[0] + items[3] / 2) * 768)
                    Ymax = int((items[1] + items[2] / 2) * 768)
                    cv2.rectangle(image, (Xmin, Ymin), (Xmax, Ymax), (255, 255, 255), thickness=2)

#hash=random.getrandbits(128)
 strr = hashlib.sha256(str(i).encode('utf-8'))
            text_hashed = strr.hexdigest()

            if i > 0:
                image = cv2.putText(image, text_hashed, (50, 50), cv2.FONT_HERSHEY_SIMPLEX, 1,
(255, 255, 255), 2,
                                    cv2.LINE_AA)

            plt.subplot(4, 4, i + 1)
            plt.imshow(image)
            plt.title("No of ships = {}".format(i))
            plt.axis('off')
```

```
            processed_images.append(image_path)
        else:
            print(f"Image not found: {image_path}")

    plt.tight_layout()
    processed_image_path = "processed_images.png"
    plt.savefig(processed_image_path)
    plt.close()

    return processed_image_path
if___name___== '_main_':
    app.run(debug=True)
```

Flask Code to Connect FrontEnd

```
from flask import Flask, request, jsonify, render_template
from full_code import process_images  # Import the function from full_code

app = Flask(_name_)
@app.route('/')
def index():
    # Render the HTML template directly from the project folder
    return render_template('front-end.html')

@app.route('/process_images', methods=['POST','GET'])
def process_images_route():
    # Access the uploaded files from the request
    uploaded_files = request.files.getlist('images')
  # Call the image processing function from full_code.py
    result_path = process_images(uploaded_files)
  # Return the relative path to the processed image
    return jsonify({'imagePath': result_path})
# Inside your Flask backend script (flask_app.py)

@app.route('/result')
def result_page():
```

```
    image_path = request.args.get('image_path', None)
    if image_path is None:
        return "Image path is missing!", 400  # Bad request
    # Directly render a template that shows the image
    return render_template('result.html', image_path=image_path)


if __name__ == '__main__':
    app.run(debug=True)
```

6. RESULT AND ANALYSIS

6.1 Classsification Report and its Confusion Matrix

A classification report shows the support, F1-score, precision, and recall for each class in a classification problem. It is a tool for performance evaluation. Confusion Matrix is a table that summarizes the true positives, false positives, true negatives, and false negatives that a classification model generates. It aids in our comprehension of our classification model's efficacy in terms of how effectively it can forecast each attack.

```
Accuracy Score is 93.7759336099585
Classification Report :
              precision    recall  f1-score   support

           0       0.94      1.00      0.97       897
           1       0.89      0.12      0.21        67

    accuracy                           0.94       964
   macro avg       0.91      0.56      0.59       964
weighted avg       0.93      0.94      0.91       964

AUC-ROC: 0.5591440789364216
LOGLOSS Value is 2.2433809163350924
```

Fig 6.1 Classification report of K-Means Algorithm

The Figure 6.1 describes the classification report of K-Means algorithm for detection of ships in the satellite images.The overall accuracy that we have achieved is 93%.

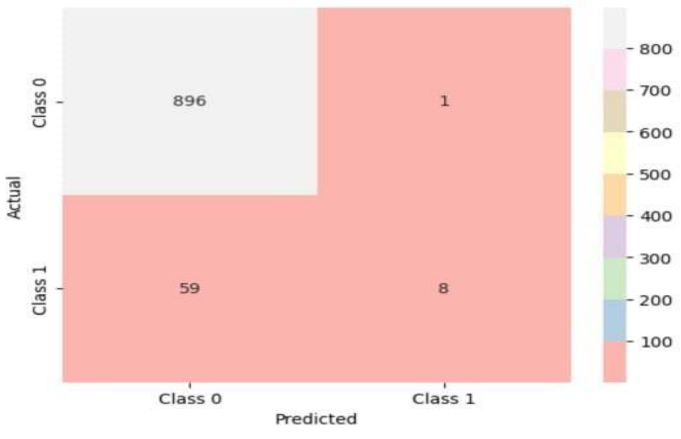

Fig 6.2 Confusion Matrix of K-Means Algorithm

The Figure 6.2 depicts the Confusion matrix of K-Means algorithm which is performed by taking only 1000 samples.

The Ships detection using satellite images is also performed by using RCNN model.The Classification report of RCNN model in shown below.

```
Accuracy Score is 94.97536945812809
Classification Report :
              precision    recall  f1-score   support

           0       0.95      1.00      0.97       957
           1       0.89      0.14      0.24        58

    accuracy                           0.95      1015
   macro avg       0.92      0.57      0.61      1015
weighted avg       0.95      0.95      0.93      1015

AUC-ROC: 0.5684430512016719
LOGLOSS Value is 1.8110604165960351
```

Fig 6.3 Classification report of RCNN model

The Figure 6.3 depicts the Classification report of RNN model to detect the ships using satellite images.The overall accuracy that we have achieved by using RCNN model is 94%.The accuracy that we have obtained for RCNN model is higher as compared with the previous K-Means model.

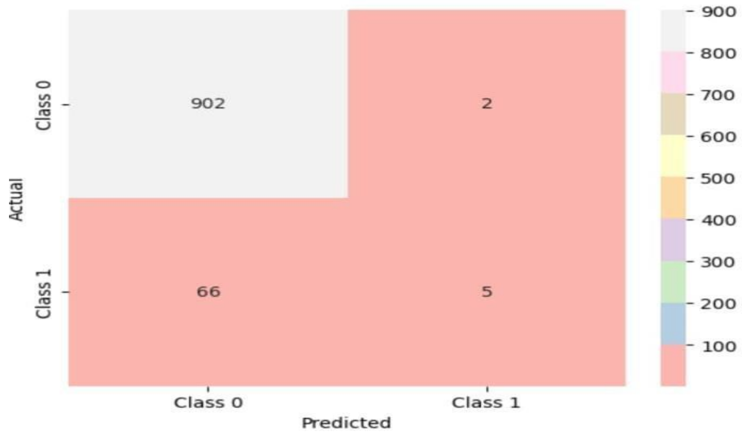

Fig 6.4 Confusion Matrix of RCNN model

The Figure 6.4 depicts the Confusion matrix of RCNN algorithm.The confusion matrix of the RCNN model likely demonstrates a high proportion of true positives and true negatives, indicating strong performance in object detection.

The third model that we used for Ships detection using satellite images is Yolov3 algorithm.Among all the three different algorithms used for object detection,it is observed that the highest accuracy is achieved for Yolov3 algorithm which is 97%.The Classification report of Yolov3 algorithm is shown below.

```
Accuracy Score is 97.95121951219512
Classification Report :
              precision    recall  f1-score   support

           0       0.98      1.00      0.99       997
           1       0.89      0.29      0.43        28

    accuracy                           0.98      1025
   macro avg       0.93      0.64      0.71      1025
weighted avg       0.98      0.98      0.97      1025

AUC-ROC: 0.6423556383436022
LOGLOSS Value is 0.738455337728254
```

Fig 6.5 Classification report of Yolov3 model

The Figure 6.5 depicts the Classification report of Yolov3 model to detect the ships using satellite images.The overall accuracy that we have achieved by using Yolov3 model is 97%.

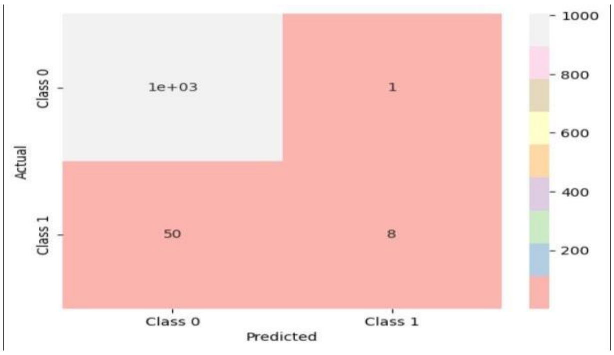

Fig 6.6 Confusion Matrix of Yolov3 model

The Figure 6.6 depicts the Confusion matrix of Yolov3 algorithm.The confusion matrix of the Yolov3 object detection.

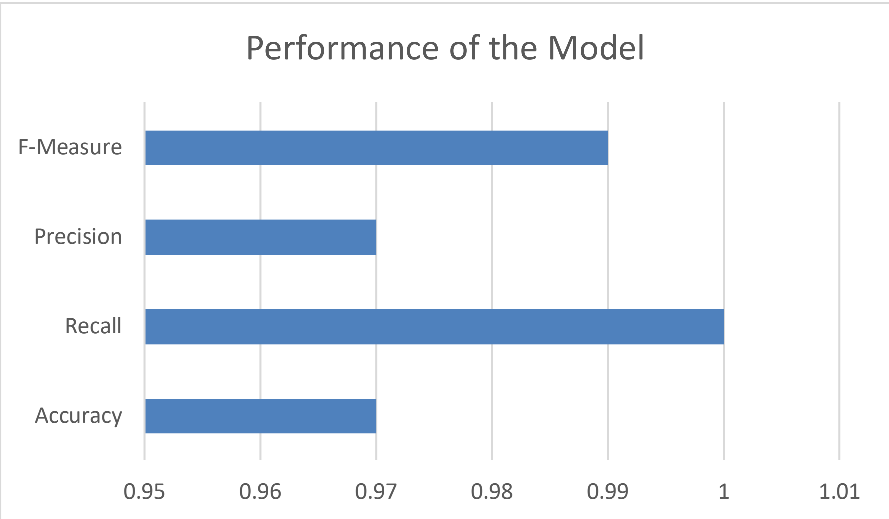

Fig 6.7 Performance of the model using various Evaluation Metrics

The Figure 6.7 visualizes how well a model performs across four key metrics: F1-score, precision, recall, and accuracy. The X-axis represents the metric values, while the Y-axis likely indicates the distribution of those values.

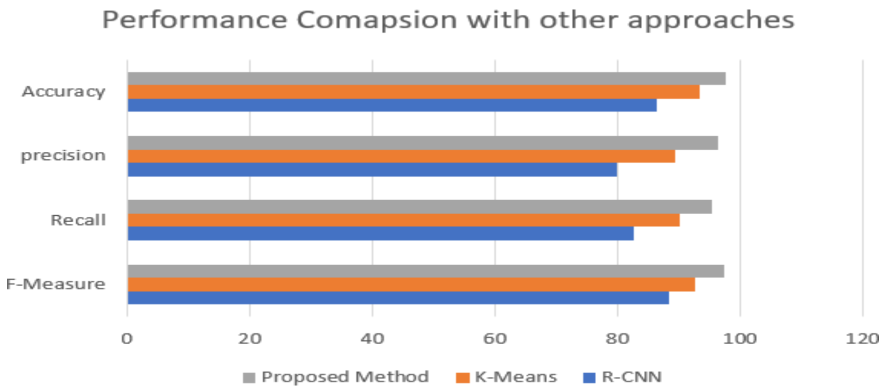

Fig 6.8 Performance Comparison with other approaches.

The Figure 6.8 depicts the performance comparison with K-Means,RCNN and Yolov3 algorithms.It is observed that the highest accuracy achieved for yolov3.

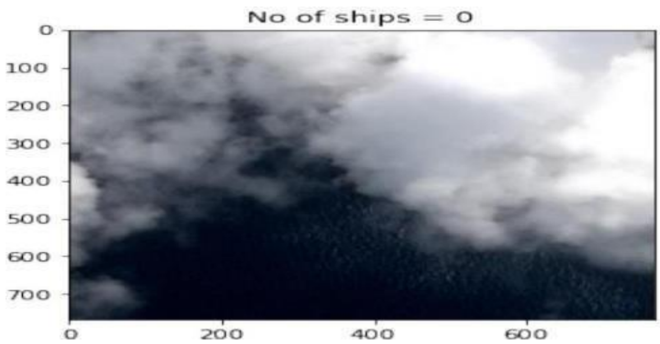

Fig 6.9. Satellite images containing no ship

Figure 6.9 shows a satellite image in which there is no ship present so that the count of the ships become zero in this case.

Fig6.10. Detection of ship from the satellite image.

Figure 6.10 shows a satellite image in which there is presence of one ship so that the result shows countof ships as one.

7. TEST CASES

Test Case 1

Input	:	Satellite images of ships
Expected Output	:	Ships detection along with ship count
Actual Output	:	Ships detction along with ship count
Result	:	Success

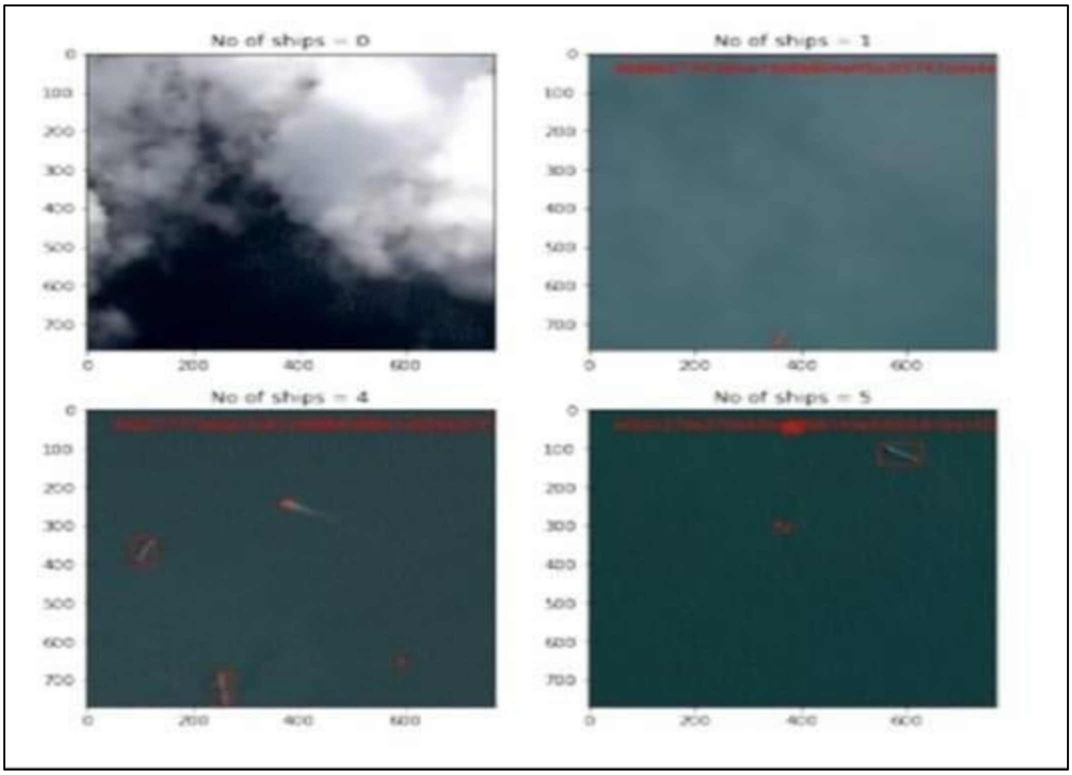

Fig 7.1 Test Case 1

The Fig 7.1 describes the output screen which contains ships present within bounding boxes and the number of ships are counted and shown as output above each image.

Test Case 2

Input	:	Satellite images of ships
Expected Output	:	Ships detection along with ship count
Actual Output	:	Please select appropriate satellite images
Result	:	Fail

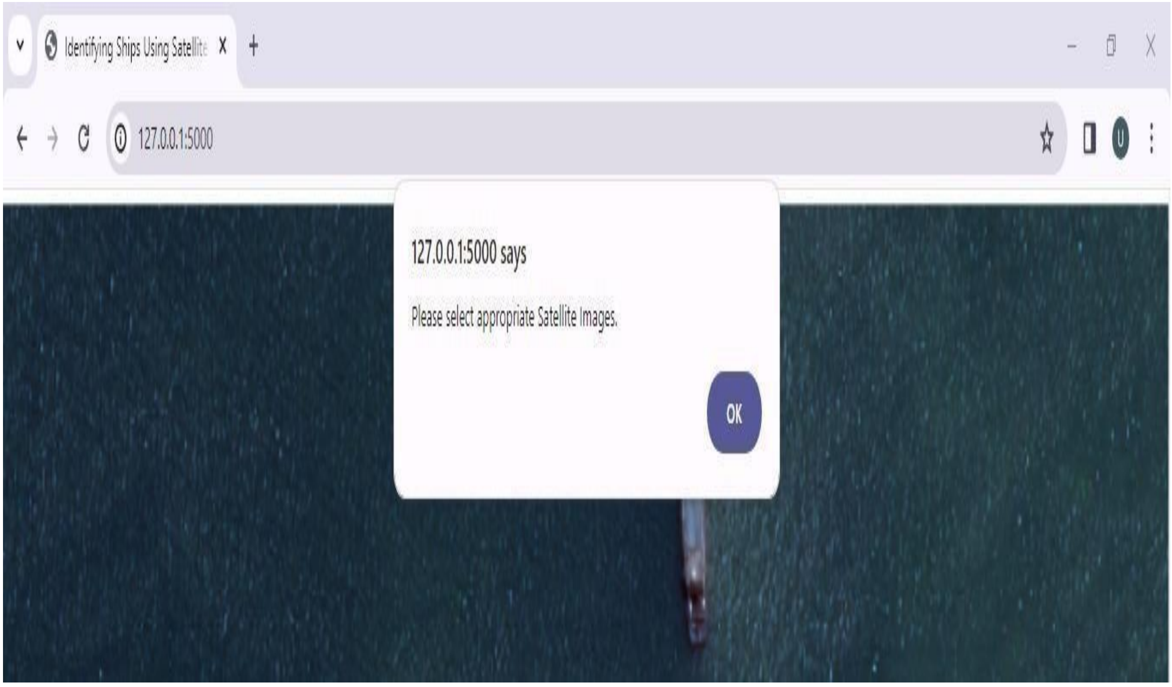

Fig 7.2 Test Case2

Figure 7.2 shows the error message that we got when we try to upload non-ship images as the input so that the user rechecks the input given to get the proper output.

8. SCREEN SHOTS

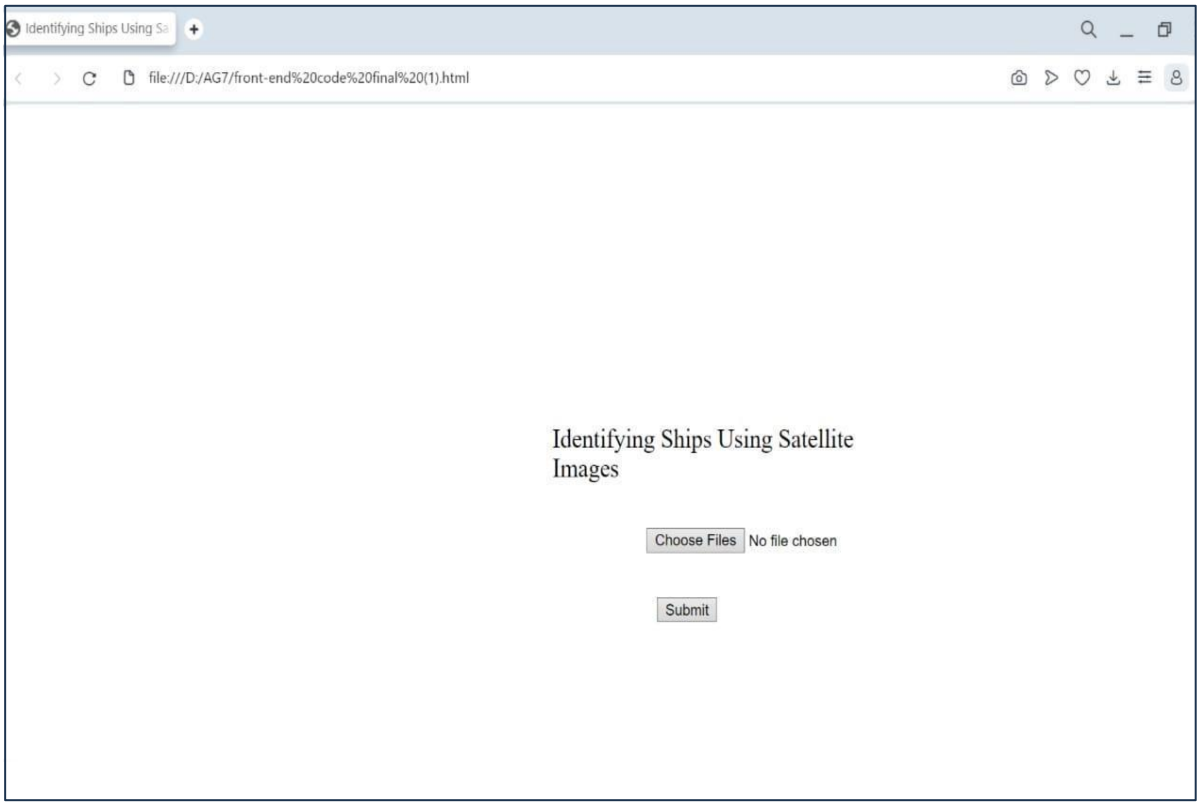

Fig 8.1Frontend screen to upload satellite images for detecting ships

Fig 8.1 depicts how to upload satellite images,click on choose files button,then click on submit to get the detectedships as output along with the count of ships present within an image.

SHIP DETECTION PAGE

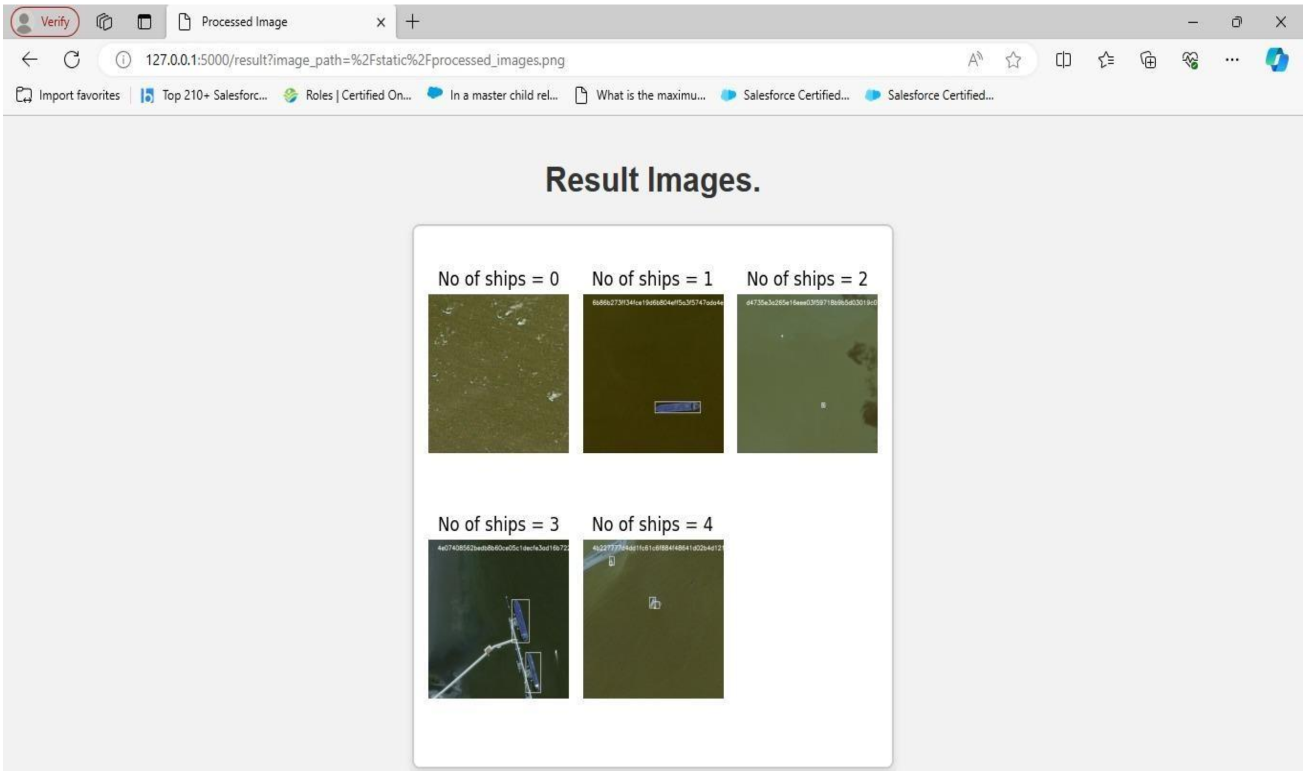

Fig 8.2 Output Screen of Ships detection in satellite images

Figure 8.2 depicts the output screen of ships detection in satellite images along with the count of ships corresponding with each image.

9. CONCLUSION

In this project,a secured, deep learning model for detecting ships is proposed. The developed model is accurate with real-time data and performs well under adverse climatic conditions due to the use of YOLOv3, which performs detection at three different layers. In addition, keen interest is taken on pre-processing the input data, as the quality and quantity of the training data deeply impact the performance and effectiveness of the model. Furthermore, the proposed bounding box function helps in accurately tying the ships and bounding boxes together. And the rel2bbox help us to identify the area wherethe ships are present which can be located by looking at the rectangular box.[14]

To improve the effectiveness and decrease the processing time, normalization and dropping those images having area less than 2 pixels was initiated. To add and push the performance of the model, downsampling of data was performed. By using this approach, more unique and import classes of data can be closely visualized rather than dealing with a plethora of data that reduce the effectiveness and increase processing time.

The main advantage of the proposed model is that it performs better in comparison to other models in case of cloud masking, land masking, and small, blurred images. Higher efficiency can be observed in case of adverse climatic conditions. Another advantage is that this model can prove equally well-performing with ships of small size, medium size, and large size that are present in the images fed tothe model.

10. FUTURE SCOPE

In the future,it is proposed to extend the model's capabilities to analyze and process video inputs as well. The incorporation of machine learning techniques, such as transfer learning and semi-supervised learning, could further improve the performance of ship detection models trained with YOLOv3. By leveraging pre-trained models on large-scale datasets, researchers can adapt these models to new environments and domain-specific challenges, enhancing their generalization capabilities and robustness in detecting ships in various maritime scenarios.

Integration of YOLOv3 with emerging satellite constellations and advanced sensors could enable real-time monitoring of maritime traffic on a global scale. Furthermore, advancements in hardware technology, such as the development of specialized accelerators for deep learning tasks, could facilitate the deployment of YOLOv3-based ship detection models on edge devices or onboard sensors. This would enable real-time analysis of video feeds directly at the source, reducing latency and bandwidth requirements for transmitting data to centralized processing units. Additionally, the integration of YOLOv3 with emerging satellite constellations and advanced sensors holds the potential to revolutionize maritime surveillance capabilities. By leveraging the extensive coverage and high-resolution imagery provided by these systems, authorities can achieve unprecedented levels of situational awareness and responsiveness in monitoring maritime activities, detecting anomalies, and enforcing maritime regulations. As these technologies continue to mature and converge, the futureof ship detection and maritime surveillance appears promising, with YOLOv3 poised to play a centralrole in driving innovation and enhancing security in maritime domains worldwide.

11. REFERENCES

[1] Nie, X. et.al, Yang, M., & Liu, R. W. Deep neural network-based robust ship detection under different weather conditions. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC) (pp.47- 52). IEEE. [October, 2019]. [https://ieeexplore.ieee.org/abstract/document/8917475]

[2] Rong, H. et.al, Ramirez-Serrano, A., Guan, L., & Gao, Y. Image object extraction based on semantic detection and improved K-means algorithm. IEEE access, 8, 171129-171139. [2020]. [https://ieeexplore.ieee.org/abstract/document/9200601]

[3] Li, H. et.al, Deng, L., Yang, C., Liu, J., & Gu, Z. Enhanced YOLO v3 tiny network for real-time ship detection from visual image. Ieee Access, 9, 16692- 16706. [2021]. [https://ieeexplore.ieee.org/abstract/document/9334973]

[4] Wang, Y. et.al, Rajesh, G., Mercilin Raajini, X., Kritika, N., Kavinkumar, A., & Shah, S. B. H. Machine learning-based ship detection and tracking using satellite images for maritime surveillance. Journal of Ambient intelligence and Smart Environments,13(5),361- 371.[2021]. [https://content.iospress.com/articles/journal-of-ambient-intelligence-and-smart-environments/ais210610]

[5] Hong et.al, Z. et.al, Yang, T., Tong, X., Zhang, Y., Jiang, S., Zhou, R., ... & Liu, S. Multi-scale ship detection from SAR and optical imagery via a more accurate YOLOv3. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, 6083-6101. [2021]. [https://ieeexplore.ieee.org/abstract/document/9448440]

[6] Gadamsetty, S. et.al, Ch, R., Ch, A., Iwendi, C., & Gadekallu, T. R. Hash-based deep learning approach for remote sensing satellite imagery detection. Water, 14(5), 707. [2022]. [https://www.mdpi.com/2073-4441/14/5/707]

[7] Nie, X. et.al, Duan, M., Ding, H., Hu, B., & Wong, E. K. Attention mask R-CNN for ship detection and segmentation from remote sensing images. Ieee Access, 8, 9325-9334. [2020]. [https://ieeexplore.ieee.org/abstract/document/8951182]

[8] Van de Sande, K. E. et.al, Uijlings, J. R., Gevers, T., & Smeulders, A. W. Segmentation as selective search for object recognition. In 2011 international conference on computer vision (pp. 1879-1886). IEEE. [November, 2011]. [https://ieeexplore.ieee.org/abstract/document/6126456]

[9] Hannevik, T. N. et.al, Olsen, Ø., Skauen, A. N., & Olsen, R. Ship detection using high resolution satellite imagery and space-based AIS. In 2010 International WaterSide Security Conference (pp. 1-6). IEEE. [November, 2010]. [https://ieeexplore.ieee.org/abstract/document/5730248]

[10] Zou, L. et.al, Zhang, H., Wang, C., Wu, F., & Gu, F. Mw-acgan: Generating multiscale high-resolution SAR images for ship detection. Sensors, 20(22), 6673. [2020]. [https://www.mdpi.com/1424-8220/20/22/6673]

[11] Chen, X. et.al, Xiang, S., Liu, C. L., & Pan, C. H. Aircraft detection by deep belief nets. In 20132nd IAPR (2020). Asian Conference on Pattern Recognition (pp. 54- 58). IEEE. [November, 2013]. [https://ieeexplore.ieee.org/abstract/document/6778281]

[12] Redmon, J. et.al, Divvala, S., Girshick, R., & Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp.779-788).[2016].[https://www.cvfoundation.org/openaccess/content_cvpr_2016/html/Redmon_You_Only_Look_CVPR_2016_paper.html]

[13] Zhao, K. et.al, & Ren, X. Small aircraft detection in remote sensing images based on YOLOv3. In IOP Conference Series: Materials Science and Engineering (Vol. 533, No. 1, p. 012056). IOP Publishing. [May, 2019]. [https://iopscience.iop.org/article/10.1088/1757-899X/533/1/012056/meta]

[14] Ying et.al, R. Airport and Ship Target Detection on Satellite Images Based on YOLO V3 Network. In Proceedings of the 6th China High Resolution Earth Observation Conference (CHREOC 2019) (pp. 167- 174). Springer Singapore. [2020]. [https://link.springer.com/chapter/10.1007/978- 981-15-3947-3_12]

[15] Nie, X. et.al, Liu, W., & Wu, W. Ship detection based on enhanced YOLOv3 under complex environments. Journal of Computer Applications, 40(9), 2561. [2020]. [http://www.joca.cn/EN/10.11772/j.issn.1001-9081.2020010097]

[16] Chang, L et.al., Chen, Y. T., Hung, M. H., Wang, J. H., & Chang, Y. L. Yolov3 based ship detection in visible and infrared images. In 2021 IEEE International Geoscience and Remote Sensing Symposium.IGARSS(pp.35493552).IEEE.[July,2021].[https://ieeexplore.ieee.org/abstract/document/9555032]

[17] Zhu, M. et.al, Hu, G., Zhou, H., Lu, C., Zhang, Y., Yue, S., & Li, Y. Rapid ship detection in SAR images based on YOLOv3. In 2020 5th international conference on communication, image and signalprocessing.

[18] Wu, T. D. et.al, Wang, H. F., Hsu, P. H., Tiong, K. K., Chang, L. C., & Chang, C. H. Target Detection and Recognition in Synthetic Aperture Radar Images Using YOLO Deep Learning Methods. In 2023 International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan) (pp.593-594). IEEE. [2023, July]. [https://ieeexplore.ieee.org/abstract/document/10226736]

[19] Guo, W. et.al, Wang, X., Lu, F., & Wang, H. Research on Intelligent Transportation Based on 'you only look at once at image' algorithm: Taking the crackdown on prostitutes' nests as an example. In 2022 International Conference on Big Data, Information and Computer Network (BDICN) (pp. 231-234). IEEE. [2022, January]. [https://ieeexplore.ieee.org/abstract/document/9758498]

[20] Bhattacharya, S. et.al, Maddikunta, P. K. R., Hakak, S., Khan, W. Z., Bashir, A. K., Jolfaei, A., & Tariq, U. Antlion re-sampling based deep neural network model for classification of imbalanced multimodal stroke dataset. Multimedia Tools and Applications, 1-25. [2020]. [https://link.springer.com/article/10.1007/s11042-020-09988-y]

[21] Gadekallu, T. R. et.al, Alazab, M., Kaluri, R., Maddikunta, P. K. R., Bhattacharya, S., & Lakshmanna, K. Hand gesture classification using a novel CNN-crow search algorithm. Complex & Intelligent Systems, 7, 1855-1868. [2021]. [https://link.springer.com/article/10.1007/s40747-021- 00324-x]

[22] Srinivasu, P. N. et.al, Bhoi, A. K., Jhaveri, R. H., Reddy, G. T., & Bilal, M. Probabilistic deep Q network for real-time path planning in censorious robotic procedures using force sensors. Journal of Real-Time Image Processing, 18(5), 1773-1785. [2021].[https://link.springer.com/article/10.1007/s11554-021-01122-x]

[23] Qi, L. et.al, Li, B., Chen, L., Wang, W., Dong, L., Jia, X., ... & Wang, D. Ship target detection algorithm based on improved faster R-CNN. Electronics, 8(9), 959. [2019]. [https://www.mdpi.com/2079-9292/8/9/959]

[24] Bhonge, H. N. et.al, Ambat, M. K., & Chandavarkar, B. R. An experimental evaluation of SHA-512 for different modes of operation. In 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-6). IEEE. [2020, July].

[https://www.researchgate.net/publication/346942868_An_Experimental_Evaluation_of_SHA-512_for_Different_Modes_of_Operation]

Identifing Ships Using Satellites Images by Using Deep Learning Techniques

Dr. M.Sireesha , M.Tech., Ph.D
Assoc.Professor,Computer Science and Engineering
Narasaraopeta Engineering College
Narasaraopet, India
moturisireesha@gmail.com

Tadavarthi Rukmini Gayathri
Computer Science and Engineering
Narasaraopeta Engineering College
Narsaraopeta
rukminigayathri3@gmail.com

Meda Asritha
Computer Science and Engineering
Narasaraopeta Engineering College
Narsaraopeta
meda.asritha@gmail.com

Bandi Saiharsha
Computer Science and Engineering
Narasaraopeta Engineering College
Narasaraopeta
Saiharshabandi90@gmail.com

***Abstract*---When using remote sensing pictures for marine security, ship detection is essential. The deep learning method for identifying ships from satellite photos is covered in this research. In order to achieve integrity Hashing is included. This model makes use of a supervised method for classifying images, and then use You Only Look Once version 3 (YOLOv3) for object recognition and feature extraction from deep CNN. By using SHA-256 the ship count will be displayed.**

KEYWORDS: remote sensing, deep learning, ship detection, YOLO v3, rel2bbox, SHA-256.

I INTRODUCTION

When compared with machines, Humans can identify objects however they are placed irrespective of size, shape and colour, while making machines to do same work requires a lot of energy and work. The YOLOV3 ML algorithm uses Deep CNN to detect objects located in the images. [12],[15]. Rel2bbox-The rel2bbox function is used to define the bounding boxes which helps us to spot the ships. It usually takes two parameters. The process is done pixel Co-ordinate, Width and Height respectively.

Satellite images are usually made up of large no of pixel which ranges from centimetres to meters. These Images are in the form of water vapour, Infrared, Visible. The modern YOLOV3 can identifies objects dynamically like in images, videos. And it underwent trained and tested on large data set .we use SHA-512 which maintains security for the transmission of data.

In this model I have used Kaggle's ship detection dataset which contains 2,30,000 photos .

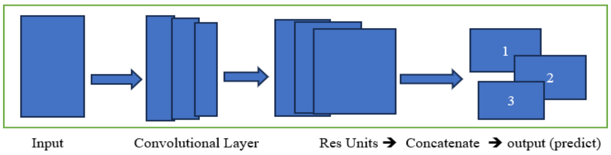

Fig. 1. Process of how YOLOV3 detects objects in Images

II LITERATRE SURVEY

The following are the references that are taken for reference for this research paper as a part of literature survey. From these manuscripts we are ableto understand how an algorithm works and what algorithm needs to be used for the identification and structuring of bounding boxes.

Jeff Faudi et.al. [1] described a methodology where the identification of ships. The help of window- shielding method the input images which are large are broken down into small images. It can only identify one kind of ship. With the help of multilevel feature extraction, the accuracy increases as it helps to even detect small objects. The limitation is that (i) it cannot detect ship which are between sea and land (ii) if the image is too small then it will increase complexity. [11]

Redmon, J. et.al [2] implemented where the system works with the dpm and uses the window shield for the detection of objects. This method uses yolo to identify the images from a single convolutional layer from that the segmentation of the images is defined and with one look it can trace where the object is present and the imperfection is that it can provide security for the location identified by the model.

Hannennvik, T. N. [3] implemented the utilization of SAR images from radarsat-2 which helps us to identify how the identification is done and the analysis, reporting of the images is combined. It has divided the ships into two categories like class A and class B, when compared with class B class A ships are easily identified and it is vulnerable – detection probability is based on the time rate of the signals, does not work well if the ships are in land masking. [9]

Van de Sande, K. E. et.al [4] used the segmentation for searching images. As we know that if we know the location of object then only, we can identify where the object is present in that image. Here he has used the segmentation to identify different locations of the object so that out of them the model can choose the best cluster from the group of locations[8]

Nie Xin et al. [5] described a model where the automatic ship detection and counting of ships will be done and with the help of yolo the process is designed by adjusting and optimizing the parameters. It can be done with the images which are HSV colour histography and LBP target features the object detection and segmentation is done. [17]

III PROPOSED SYSTEM

Our proposed system for identification of ships mainly consists of the following steps:

(A) Dataset Preprocess
(B) Bounding Boxes
(C) Normalization
(D) Ship Count
(E) Bounding Boxes.
(F) Hashing Value.

(A) Dataset Preprocess: is to load the data set which is Kaggle air bus ship detection and analyze the data set. With the help of pandas library, we have read the data set in which we came to know that the data set consists of 2 columns namely Image Id and Encoded Pixels.

(B) Bounding Boxes: To define the bounding boxes the bounding boxes can be defined by the help of rel2bbox function which takes two parameters those are rel and shape and it returns four values which are X coordinate, Y co-ordinate which defines the shape of the box, Width, Height.

(C) Normalization: To Normalize and drop - encode the bounding boxes. After defining the bounding boxes now we need to normalize the encoded pixels and the images which are normalized with the help of drop function from pandas we will remove the encoded pixels which has no bounding boxes the normalization.

(D) Ship Count: This involves the visualization of the count of ships this is carried out as, first let us take a variable as count at first it is initialized to 0 and if the image has bounding boxes, then the count is going to be incremented by one and cross filter is going to be applied in order to protect the integrity of the algorithm. With the help of format specifier, the count of the ships is going to be visualized on top of the images.

(E) Bounding Boxes:This to visualize the bounding boxes as we have already defined the bounding boxes now the function is going to be called and with the help of encoded pixel values the bounding box is going to be drawn and with the help of this, we will be able to identify the ships.

(F) Hashing Values *:*The involves the hashing of the location of ships along with the count of ships. In order to provide the encoded value to these we have used SHA-512 function from the hash library. And it uses 512 bits and these are divided into 64 bytes and this encoded algorithm.

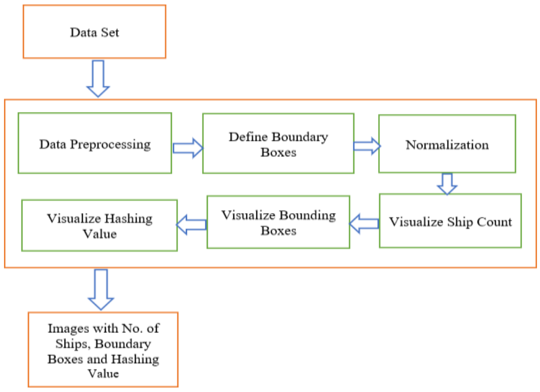

Fig .2. Proposed Methodology Architecture

Now we need find the find the area of the bounding boxes sothat the area which is less than 1 percentile is removed and with the help of this we will able to identify the majority of ships size.

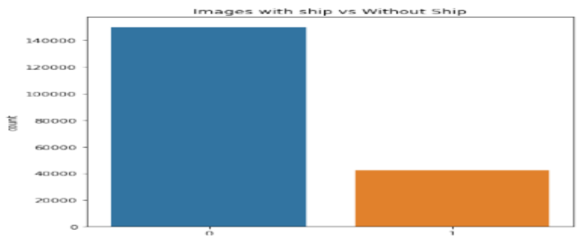

Fig .3.Images with Ship VS Without Ship

X-axis: bounding box width

Y-axis: bounding box height

Now we have identified how many ships are there in a image. Like the no. of images with 0 ships and the no. of images with 1 ship etc..

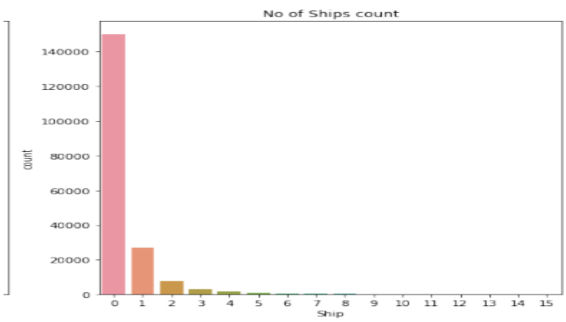

Fig .4. Number of ships in a image

X-axis: bounding box width

Y-axis: bounding box height

The Fig.4 shows the images that have at least one image. Here we have tried to find out whether satellite images have at least one image.

If there is no image then we are trying to remove the data of images which don't have ships in them so that data gets simplified.

PERFORMANCE ANALYSIS

Confusion Matrix:

Confusion matrix helps us to understand how well our model performs on the data set.This shows the count of accurate and inaccurate model based on actual outcome.

It is primarily made up of four distinct components:

True-Positives: are instances in which the model accurately forecasts a positive class, such as when the ship is present and detected.

True-Negatives: when the model accurately forecasts a negative class in the absence of the ship

False-Positives: These occur when the model forecasts the positive class inaccurately, even when the ship is actually present.

False-Negatives: When the ship is absent while the model predicts it to be present, this leads to an inaccurate negative class prediction.

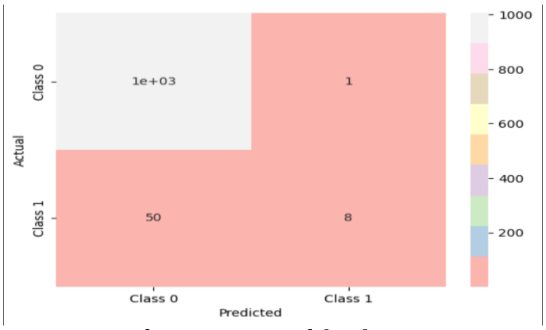

Fig .5. Confusion Matrix of the data set

While building this model we have also compared with other approaches like R-CNN and K-means in means of accuracy, precision, F1-Score, Recall out which if R-CNN gives an overall accuracy of 89% and K-Means g

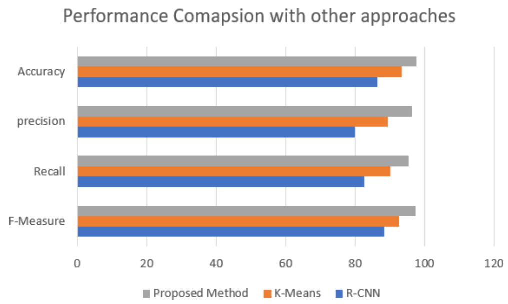

Fig.6.Performance Comparison with other approaches.

ACCURACY:

Accuracy is used to calculate the performance of a algorithm. Classified instances are measured correctly among all the instances in the test dataset.

From the confusion matrix when can calculate the accuracy it can be done by the formula of sum of true-positive (N_{TP}) and true-negative (N_{TN}) divided by sum of true-positive (N_{TP}), true-negative (N_{TN}), false- positive (N_{FP}) and false-negative (N_{FN}).

$$\text{Accuracy} = \frac{(N_{TP}+N_{TN})}{(N_{TP}+N_{TN}+N_{FP}+N_{FN})}$$

Recall:

Recall usually know as sensitivity. It is the measure of correctly identifying the true positives, or predicting correctly from the true positive cases. IT can be calculated by using following formula,

$$\text{Recall} = \frac{N_{TP}}{(N_{TP}+N_{FN})}$$

Precision:

Precision states that out of all the positive identification what are actually correct or at what level of the data that is predicted to be positive are actually positive. It is evaluated by the formula.

$$\text{Precision} = \frac{N_{TP}}{(N_{TP}+N_{FP})}$$

F1-Score:

Its primary application is model comparison. F1Score makes use of both precision and recall. Harmonic mean is taken into account rather than arithmetic mean. Assuming that R-recall and P-precision, the formula below can be used to determine the F1Score:

$$\text{F1-Score} = \frac{2*R*P}{(R+P)}$$

This performance in the model is shown in the diagram below when compared to other models. It provides high accuracy and operates effectively when a sizable data set is provided as an input to the algorithm.

```
Accuracy Score is 97.3076923076923
Classification Report :
              precision    recall  f1-score

           0       0.97      1.00      0.99
           1       0.78      0.21      0.33

    accuracy                           0.97
```

Fig.7. Performance of the model.

Result:

Fig.8. describes the scenario 1, the satellite image does not have any ships in them, so the output is displayed as below, there are 0 ships so count function displays the value of 0. As there is no ship that's why the hashing does not work.

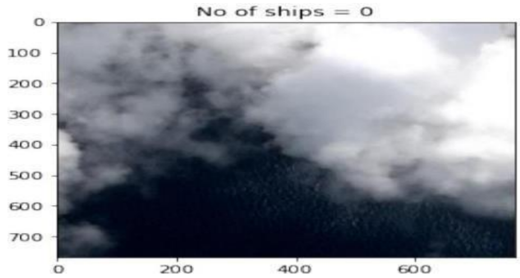

Fig. 8. No ships in the satellite image

The shape of the image which we want as an output the function returns the value of X-axis, Y-axis, breadth and height with the bounding box is going to be drawn on the image which helps us to spot the ship on the image.

Fig.9. Detection of ship from the satellite image.

Next step we have tried to count the no of ships that are in the image. The algorithm is further extended to count ships that are in the image.

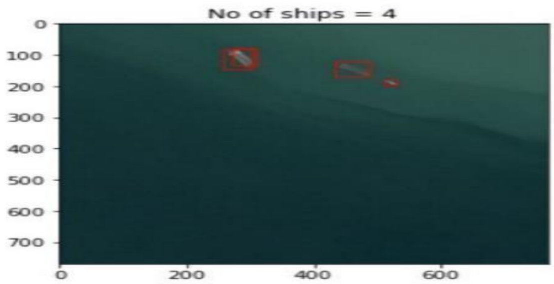

Fig.10.How the ships are counted and displayed on the top of the image.

One of the safest and most difficult to crack algorithms, the SHA-512 algorithm, is used to generate the hashing value. Using a brute force strategy, it would take 2^256 time to decode the hash value without the key.

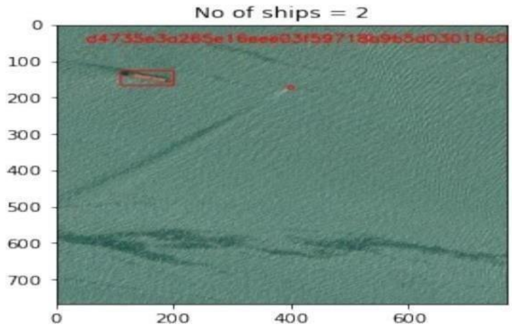

Fig.11. Hashing Value along with Count of ships

X-axis: bounding box width

Y-axis: bounding box height

The implementation of hashing is done numerous real time applications, where locating missing ships and preventing trafficking of illicit commodities.

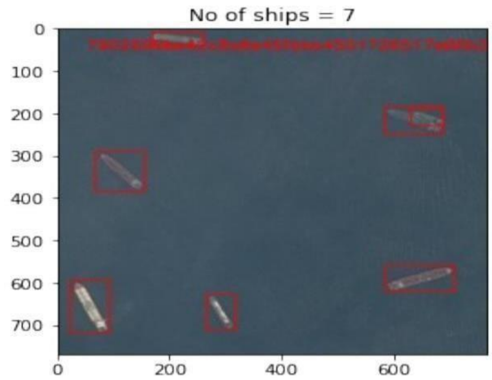

Fig.12. Another picture hashing Value along with Count of ships.

X-axis: bounding box width

Y-axis: bounding box height

Since hashing is unique and irreversible, it is desirable to utilize it with the model. Since all of the SAR data originates from satellites and is initially centralized, it might become a potential target and be readily attacked if security mechanisms are not implemented to safeguard it. Thus, to guarantee secure data, effective algorithms are employed.

IV CONCLUSION AND FUTURE SCOPE

In the paper a secure enhanced way to identifying of ships using the deep learning approach hasbeen done, the data which we have performed operationsis all real time data and with this we can say that it worksaccurately for any type of images like cloud, land masked and high resolution, over any climatic data.

By the utilization of YOLOV3 algorithm and its multilevel feature extraction and identification helps us to detect theimages at three levels which are small, medium and large. And the rel2bbox help us to identify the area where the ships are present which can be located by looking atthe rectangular box.

Normalization has been done in order to remove the duplicate values and also for the bounding box area which less than 1 percent so that the accuracy and effectiveness is improved and processed time also reduced.

Furthermore, this model is also works for the user data if the user gives the input of images and the backend code will process and gives the output as it is required. The usage of SHA-512 algorithm gives the integrity and security to the loc of the ship and count of the ships so that they remain confidential.

V REFERENCES

[1] Nie, X. et.al, Yang, M., & Liu, R. W. Deep neural network-based robust ship detection under different weather conditions. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC) (pp.47- 52).IEEE. (2019, October).[https://ieeexplore.ieee.org/abstract/document/8917475]

[2] Rong, H. et.al, Ramirez-Serrano, A., Guan, L., & Gao, Y. Image object extraction based on semantic detection and improved K-means algorithm. IEEE access, 8171129171139.(2020)[https://ieeexplore.ieee.org/abstract/document/9200601]

[3] Li, H. et.al , Deng, L., Yang, C., Liu, J., & Gu, Z. Enhanced YOLO v3 tiny network for real-time ship detection from visual image. Ieee Access, 9,16692- 16706 (2021)[https://ieeexplore.ieee.org/abstract/document/9334973]

[4] Wang, Y. et.al Rajesh, G., Mercilin Raajini, X., Kritika, N., Kavinkumar, A., & Shah, S. B. H. Machine learning-based ship detection and tracking using satellite images for maritime surveillance. Journal of Ambient Intelligence and Smart Environments, 13(5),361371.(2021)[https://content.iospress.com/articles/journal-of-ambient-intelligence-and-senvironments/ais210610]

[5] Hong et.al, Z. et.al, Yang, T., Tong, X., Zhang, Y., Jiang, S., Zhou, R., ... & Liu, S. Multi-scale ship detection from SAR and optical imagery via a more accurate YOLOv3. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, 6083-6101.(2021). [https://ieeexplore.ieee.org/abstract/document/9448440]

[6] Gadamsetty, S. et.al, Ch, R., Ch, A., Iwendi, C., & Gadekallu, T. R. Hash-based deep learning approach for remote sensing satellite imagery detection. Water, 14(5), 707. (2022)[https://www.mdpi.com/2073-4441/14/5/707]

[7] Van de Sande, K. E. et.al , Uijlings, J. R., Gevers, T., & Smeulders, A. W. Segmentation as selective search for object recognition. In 2011 international conference on computervision(pp.18791886).IEEE.(2011,November).

[https://ieeexplore.ieee.org/abstract/document/6126456]

[8] Hannevik, T. N. et.al , Olsen, Ø., Skauen, A. N., & Olsen, R. Ship detection using high resolution satellite imagery and space-based AIS. In 2010 International Water Side Security BY Conference (pp.1-6). IEEE.(2010,November). [https://ieeexplore.ieee.org/abstract/document/5730 248]

[9] Zou, L. et.al, Zhang, H., Wang, C., Wu, F., & Gu, F. Mw-acgan: Generating multiscale high-resolution SAR images for ship detection. Sensors, 20(22), 6673.

[10] Chen, X. et.al,Xiang, S., Liu, C. L., & Pan, C. H. Aircraft detection by deep belief nets. In 2013 2nd IAPR (2020). [https://www.mdpi.com/1424-8220/20/22/6673]

Asian Conference on Pattern Recognition (pp. 54-58).IEEE.(2013,November).[https://ieeexplore.ieee.org/abstract/document/6778281]

AG7 conference paper final.pdf

ORIGINALITY REPORT

4%	2%	4%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Mariam Ahmed Sheha, Amr Sharwy, Mai S. Mabrouk. "Pigmented skin lesion diagnosis using geometric and chromatic features", 2014 Cairo International Biomedical Engineering Conference (CIBEC), 2014 Publication	1%
2	www.mdpi.com Internet Source	1%
3	qmro.qmul.ac.uk Internet Source	1%
4	"Advances in Machine Learning and Computational Intelligence", Springer Science and Business Media LLC, 2021 Publication	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		

NARASARAOPETA ENGINEERING COLLEGE (AUTONOMOUS)

Approved by AICTE, Permanently Affiliated to JNTUK, Kakinada, NIRF Ranking (251-300 Band), Accredited by NBA (Tier-I) & NAAC with 'A+' Grade
Kotappakonda Road, Yellamanda (Post), Narasaraopet - 522601, Palnadu Dist., Andhra Pradesh, INDIA. Website:www.nrtec.in

International Conference on

Paper ID
NECICAIEA-2K24-133

Artificial Intelligence and Its Emerging Areas

NEC-ICAIEA-2K24

12^{th} & 13^{th} April, 2024

Organized by Departments of CSE, IT, CSE(AI), CSE(AI&ML), CSE(DS), CSE(CS) & MCA in Association with CSI

Certificate of Presentation

This is to Certify that Tadavarthi Rukmini Gayathri, Narasaraopeta Engineering College has presented the paper title **Identifying Ships Using Satellite Images** in the International Conference on Artificial Intelligence and Its Emerging Areas-2K24 [NEC-ICAIEA-2K24], Organized by Department of Computer Science and Engineering, CSE(AI),IT,CSE(AIML),CSE(DS),CSE(CS) and MCA in Association with CSI on 12^{th} and 13^{th} April 2024 at NARASARAOPETA ENGINEERING COLLEGE (AUTONOMOUS), Narasaraopet, A.P., India.

Convenor	Chief-Convenor	Principal, Patron
Dr.S.V.N.Srinivasu	Dr.S.N.Tirumala Rao	Dr. M. Sreenivasa Kumar

MHRD'S INNOVATION CELL (GOVERNMENT OF INDIA)

NARASARAOPETA ENGINEERING COLLEGE (AUTONOMOUS)

Approved by AICTE, Permanently Affiliated to JNTUK, Kakinada, NIRF Ranking (251-300 Band), Accredited by NBA (Tier-I) & NAAC with 'A+' Grade
Kotappakonda Road, Yellamanda (Post), Narasaraopet - 522601, Palnadu Dist., Andhra Pradesh, INDIA. Website:www.nrtec.in

International Conference on

Paper ID
NECICAIEA-2K24-133

Artificial Intelligence and Its Emerging Areas

NEC-ICAIEA-2K24

12th & 13th April, 2024

Organized by Departments of CSE, IT, CSE(AI), CSE(AI&ML), CSE(DS), CSE(CS) & MCA in Association with CSI

Certificate of Presentation

This is to Certify that Meda Asritha, Narasaraopeta Engineering College has presented the paper title **Identifying Ships Using Satellite Images** in the International Conference on Artificial Intelligence and Its Emerging Areas-2K24 [NEC-ICAIEA-2K24], Organized by Department of Computer Science and Engineering, CSE(AI),IT,CSE(AIML),CSE(DS),CSE(CS) and MCA in Association with CSI on 12th and 13th April 2024 at NARASARAOPETA ENGINEERING COLLEGE (AUTONOMOUS), Narasaraopet, A.P., India.

Convenor	Chief-Convenor	Principal, Patron
Dr.S.V.N.Srinivasu	Dr.S.N.Tirumala Rao	Dr. M. Sreenivasa Kumar

NARASARAOPETA ENGINEERING COLLEGE (AUTONOMOUS)

Approved by AICTE, Permanently Affiliated to JNTUK, Kakinada, NIRF Ranking (251-300 Band), Accredited by NBA (Tier-I) & NAAC with 'A+' Grade
Kotappakonda Road, Yellamanda (Post), Narasaraopet - 522601, Palnadu Dist., Andhra Pradesh, INDIA. Website:www.nrtec.in

Paper ID
NECICAIEA-2K24-133

International Conference on
Artificial Intelligence and Its Emerging Areas
NEC-ICAIEA-2K24
12th & 13th April, 2024
Organized by Departments of CSE, IT, CSE(AI), CSE(AI&ML), CSE(DS), CSE(CS) & MCA in Association with CSI

Certificate of Presentation

This is to Certify that Bandi Sai Harsha, Narasaraopeta Engineering College has presented the paper title **Identifying Ships Using Satellite Images** in the International Conference on Artificial Intelligence and Its Emerging Areas-2K24 [NEC-ICAIEA-2K24], Organized by Department of Computer Science and Engineering, CSE(AI),IT,CSE(AIML),CSE(DS),CSE(CS) and MCA in Association with CSI on 12th and 13th April 2024 at NARASARAOPETA ENGINEERING COLLEGE (AUTONOMOUS), Narasaraopet, A.P., India.

Convenor	Chief-Convenor	Principal, Patron
Dr.S.V.N.Srinivasu	Dr.S.N.Tirumala Rao	Dr. M. Sreenivasa Kumar