Mehran University Research Journal of Engineering and Technology

https://doi.org/10.22581/muet1982.3186

2024, 43(2) 205-219

Enhancing object detection and classification using deep learning: A novel approach with annotated dataset and YOLO-C3GTR architecture

Rana Muhammad Usman [a, *], Junhua Yan [a], Saqib Mehmood [b], Muhammad Jamil [c]

[a] *Department of Space Optoelectronic Information, College of Astronautics, Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing 211106, China*

[b] *Department of Open and Environmental Systems, Graduate School of Science and Technology, Keio University, Tokyo 223-0061, Japan*

[c] *Department of Computer Science, COMSATS University Islamabad, Sahiwal Campus, Sahiwal 57000, Pakistan*

[*] Corresponding author: Rana Muhammad Usman, Email: ranausman@nuaa.edu.cn

Received: 17 February 2024, Accepted: 28 March 2024, Published: 01 April 2024

KEYWORDS

Object Detection

Classification

Deep Learning

YOLO-C3GTR

Computer Vision

Dataset Annotation

ABSTRACT

In the rapidly evolving field of computer vision, the need for advancements in object detection is paramount. For a variety of reasons, including security, maritime surveillance, and environmental monitoring, the identification and categorization of ships in aerial pictures is an essential component. In this study, two distinctive approaches are proposed to improve the accuracy of ship detection and categorization in satellite imagery. In the beginning, the dataset for ship detection that was provided by Airbus was separated into five distinct categories: oil tanker, bulk carrier, container ship, sand carrier, and general ship class. Setting up this division was done with the idea of increasing the precision of the ship classification in the imagery. Moreover, we deployed the YOLO-C3GTR model, which is part of our Object Detection framework and helped us to distinguish ships using the technique of Object Detection. The application of the methodology that we have proposed has the potential to significantly improve the precision of these systems when they are used in real-world scenarios and to make a novel addition to the field of ship identification and classification. We have used multiple algorithms of YOLOv5 tested on this dataset like TPHYOLOv5, YOLOv5x, YOLOv5l and our proposed architecture YOLO-C3GTR and it surpassed all the existing approaches by the mAP of 0.83. Furthermore, we intend to perform additional research in this field while simultaneously working to improve our technique. A comparison was made between the performance of the model and that of other deep learning approaches. The model was evaluated and trained with the categorized dataset.

1. Introduction

Artificial intelligence methods are gaining popularity throughout the world due to their ability to provide accurate predictions and save time. In addition to being used in many fields, Artificial Intelligence/Machine Learning has been used in the medical sector [1], energy sector [2], banking sector [3], transportation sector [4]

© Mehran University of Engineering and Technology 2024

and numerous other fields. Algorithms such as deep learning and statistical methods [5], [6], LR, SVR, LSTM [7], and transformer on hybrid systems for hydrogen production from renewable energy [8] and forecasting of energy production for hybrid solar-wind systems [9] have been incorporated. As artificial intelligence integrates effortlessly with object identification[10], a unique narrative arises. Object identification is an important topic in the field of computer vision and has garnered significant academic focus over the last several decades. The detection approach necessitates localizing items using bounding boxes and accurately assigning the appropriate class to each proposition. Over the last several years, several advanced convolutional neural networks (CNNs), including SOLO [11], SSD [12], CoupleNet [13] and RefineSSD [14], have shown remarkable effectiveness in object identification. Deep learning techniques, including You Only Look Once (YOLO) [15], RCNN [16], Fast RCNN [17], Faster RCNN [18], and Mask RCNN [19], have been widely used. These algorithms have shown commendable performance across several natural picture datasets. Deep-learning algorithms have been used to recognize objects in remote-sensing photos.

Ship detection is an indispensable application in preserving marine security, where traffic control, oil spillage, and sea pollution surveillance are carried out. Utilization of ship detection supplies vital data for strategic planning. Operating as an example of object detection, it has mastered the strategies of general detection and yet it still struggles with size variation and the presence of small units. Some illustrations in Fig. 1 are the case in point.

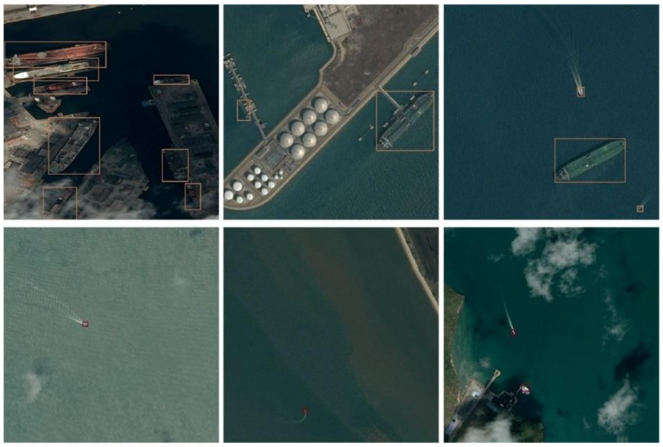


Fig. 1. Depiction Of Several Instances of Size Variation (In the First Row) And Problems Posed by Small Objects (In the Second Row)

This research focused on the identification and categorization of 5 different types of ships: oil tankers, bulk carriers, container ships, sand carriers, and other unclassified ships. The various categories of ships have unique characteristics that set them apart. Oil tankers, for example, are usually large, flat-bottomed vessels with cylindrical tanks used for transporting oil. In contrast, bulk carriers are ships designed for carrying unpackaged bulk cargo like grains, coal, and ores. Contrarily, containers are designed to be easily changed between various forms of transportation, making container ships ideal for the transportation of these standardized cargo units.

In addition, we made accurate adjustments to the Airbus Ship Detection dataset to match the ship categories we were focussing on. By manually annotating images with bounding boxes, this updated approach showed ship positions and classes. Based on this strategy, the YOLO-C3GTR model recognized and classified ships, accurately representing the required ship types.

The capability of our ship identification and classification method has a wide range of applications in the real world. An illustration of this would be assisting port authorities in the regulation of marine traffic and the identification of security threats such as piracy and smuggling. In addition to this, it can monitor ships that unlawfully dump cargo. The identification of ship categories for search and rescue operations is another way that it helps with environmental surveillance.

Ships were accurately identified and categorised using YOLO-C3GTR. YOLO uses a single network to detect and classify objects.

Contributions:

1. The Airbus Ship Detection Dataset was classified into five main classes: general ships, oil tankers, bulk carriers, sand carriers, and container ships.
2. We have deployed a YOLO-C3GTR framework that has received particular instructions to detect and classify such types of ships.
3. Our solution has the potential to contribute to the management of maritime traffic, support search and rescue missions, and facilitate environmental monitoring through the identification and tracking of ships.

© Mehran University of Engineering and Technology 2024

2. Related Work

2.1 Object Detection in Computer Vision

Object detection can be said to be a field of computer vision which is considered to be one of the most difficult jobs in the world of computer vision. In recent years deep learning researchers have shown great success with convolutional neural networks (CNNs), which have become the most favoured model for semantic segmentation [20], [21], image classification [22], [23], object detection [18], [24], and many other applications. The deep learning-based anchor-based object identification techniques are either one-stage or two-stage categories. Object detection by two-stage detection algorithms employs two different phases to accomplish the purpose of detection. The basis of the stages process is the box discovery of a preselected set or the use of a CNN network. Next, the selector boxes are sent for classification and regression operations. The two-stage process offers an important plus point in the form of highly precise outcomes. There are only two steps for single-stage object detection techniques where it is the first step that performs detection. The main concept that the one-stage approach employs is taking a variety of samples from different pixels in the image. Numerous sampling scales and aspect ratios are being employed and, afterwards, the utilization of Convolutional Neural Networks (CNN) to extract the features. One of the main advantages of using the one-stage technique is the fact that processing is extremely fast as the whole process is accomplished in one step only. On the other hand, a major disadvantage of prolonged intensive sampling is the augmented complexity of the training. Such a situation can be due to a notable deficiency between positive and negative examples, which undermines the slight accuracy of the model.

The R-CNN family is the most famous two-step object detection method. R-CNN is an original object detection technique using CNN, which first creates regional suggestions from selective search [16]. The CNN model receives the previous ones and classifies them according to their groups. One of the main modifications made in the Fast R-CNN [17] is a more efficient version of R-CNN [16], that further improves object detection performance. Contrasting to the traditional feature extraction process in which global features are extracted from whole original input photos, our proposed method allows for extracting features from each regional proposal. Faster R-CNN [18] is a later upgrade of Fast R-CNN, distinguished by several novel contributions. The Faster R-CNN design is based on the regional proposal network (RPN) that proposes objects by itself, hence allowing us to purely implement object detection tasks using neural networks. In the same turn, by employing Reverse Polish Notation (RPN), the load of computations is considerably lightened down and the speed of calculations is noticeably elevated. Mask R-CNN [15], [19] is a keener variant of Faster RCNN which uses ROI-Align to generate more precise abstraction for each proposal. Concurrently, it completes a different branch of work doing the segment task. Among these approaches, one can mention the most widely used techniques in single-stage object identification include YOLO [15] and SSD [12]. The algorithms of YOLO v4 [25] actively identify the dispersion of the bounding box in comparison with the box that precedes it which implies that the training process will be much more convenient. YOLO v4 employs multi-scale prediction to detect and produce the results whether the object is small or large. As opposed to You Only Look Once (YOLO), SSD achieves detection by implementing Convolutional Neural Networks that do not require an extra classifier to predict the target after the final layer.

The aforementioned methods exemplify the contemporary state of the object identification domain, signifying that the current object identification technique has reached a highly developed level.

2.2 Ship Detection in Remote Sensing Images

Ship detection techniques now in use are often enhanced iterations of traditional object identification algorithms. Ref. [26] analysed the attributes of satellite images and enhanced the detection accuracy of YOLO v2 [27] for identifying objects in distant sensing. Tang et al. [28] used wavelet coefficients derived from the compressed domain of JPEG2000, together with a combination of deep neural network (DNN) and extreme learning machine (ELM), to address the ship identification issue in remote sensing photos. Zou et al. [29] introduced SVD Net, a model that combines CNN with the singular value decomposition (SVD) technique. Li et al. [30] included hierarchical selective filtering layers onto Faster R-CNN [18] to address the challenges arising from varying ship sizes. Yang et al. [31] introduced a ship detection system that used saliency segmentation and the local binary pattern (LBP) descriptor to differentiate ship characteristics. The R-CNN architecture incorporates a unique layer that is capable of maintaining rotation invariance, specifically designed to address the issue of objects with varying orientations

© Mehran University of Engineering and Technology 2024

[32]. Liu et al. [33] enhanced the ship detection accuracy by modifying the dimensions of anchor boxes, estimating the localization uncertainties of bounding boxes, using soft non-maximum suppression, and rebuilding a combined loss function. Zhao et al. [34] introduced the attention-receptive pyramid network (ARPN) as a solution for detecting ships at many scales. Nevertheless, the current ship recognition methods may not possess the capability to address the challenge of detecting several types of ships with little training data in remote sensing photos.

3. Methodology

3.1 Dataset Description

In 2018, Airbus Defense and Space made the Airbus Ship Detection Dataset available. This is a comprehensive collection of satellite photos specifically designed for ship detection purposes. This system comprises about 2 million image chips. Every individual is 256 by 256 pixels in size and occupies an approximate area of 0.065 square kilometres. These images were acquired by the WorldView-2 satellite, which is capable of distinguishing features as small as 0.5 meters. The expansive assemblage of aquatic features spans an area of over 1.4 million square kilometres. The information was gathered under various meteorological and illumination circumstances.

Table 1

Dataset Description of Airbus Ship Detection Challenge

Train Images	192,559
Test Images	15,606
Dimension	768x768 pixels
Horizontal Resolution	96dpi
Vertical Resolution	96dpi
Bit Depth	24

The dataset consists of pixelated images labelled to indicate whether a ship is present or not. This classification facilitates the dataset's utilization in tasks related to object detection. Additionally, the collection provides coordinates that enable tracking of vessels. Airbus offers a range of free resources and methods for acquiring and utilizing the data for non-commercial purposes. However, an Airbus license is mandatory for any industrial application.

The Airbus Ship Identification Dataset which is relies very much on researchers and developers for the development and refinement of automatic ship identification systems. For the generation of this dataset, the "You Only Look Once" (YOLO) neural network was selected, which is state-of-the-art for the object detection task. The main function of this network model is to forecast object clustering margins and the probability of inclusion utilizing a single neural network. YOLO is considered outstanding for its high speed and accuracy, which makes it the optimal choice for real-time applications, most notably monitoring marine and maritime traffic. The dataset then provides a rich variety of visual cues and performs competently in ship identification, which makes it a fine resource for the development and evaluation of the algorithm. High accuracy and widespread applicability increase the role of this instrument in the diagnostic process.

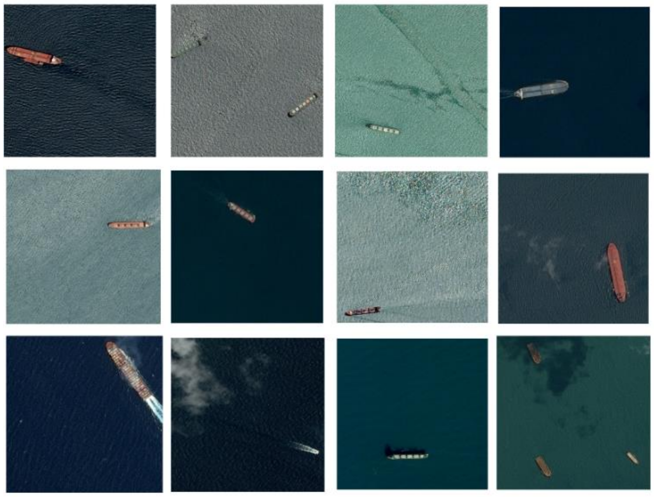


Fig. 2 Airbus Ship Detection Dataset

3.2 Dataset Annotation for Object Detection Technique

The tool that is mostly used to annotate the images is makesense.ai. To make ship images be classified into five groups, all of the images were self-annotated as presented in Fig. 3. We are excited about the findings of the new technique, and we proved that the technique is very effective in determining and also classifying satellite images.

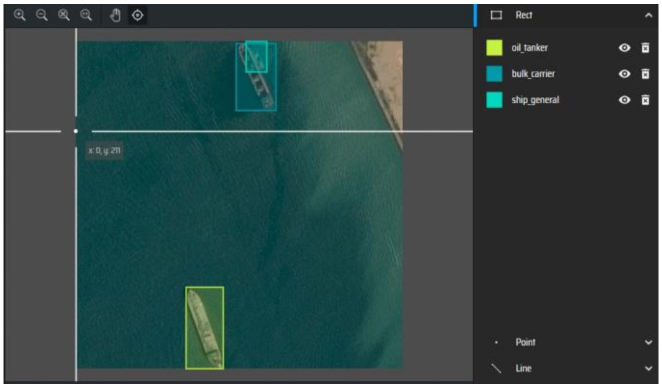


Fig. 3 Annotation of Ship Detection Dataset

With great care and attention to detail, we expertly annotated a total of two thousand images, putting into practice all of the previously developed methodologies. The technique consisted of completing a thorough study of each image to identify certain kinds of ships, properly drawing the limits of those ships, and classifying them according to the relevant categories based on those boundaries.

We have done a thorough analysis of the characteristics of each image to ensure the accuracy of the annotations. This analysis includes several factors like sightline, perspective and lighting. Moreover, we have made efforts to minimize the possibility of errors. We have achieved this goal by conducting each annotation carefully and by making necessary modifications.

3.3 Dataset Classification

The collection of datasets is divided into five distinct kinds of ships, which are as follows: oil tankers, bulk carriers, container ships, carriers, and general ships (also known as unclassified ships):

3.3.1 Oil tankers

These ships are purposefully designed for the conveyance of oil or petroleum products and are frequently referred to as huge cargo ships.

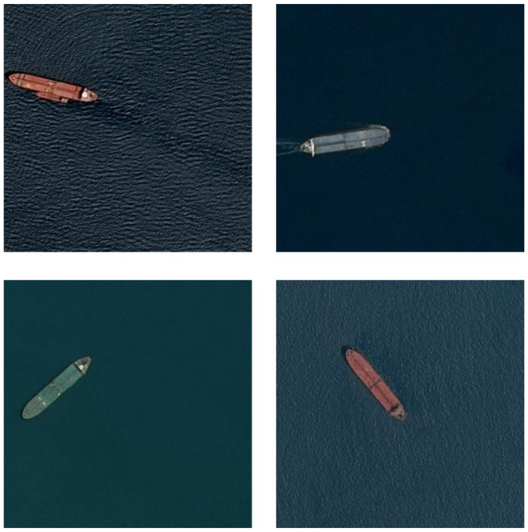


Fig. 4. Oil Tankers

These colossal vessels are characterized by their tremendous scale, commonly spanning somewhere between 100,000 to 320,000 deadweight tonnage, and by hulls that either assume a cylindrical or bulbous form.

3.3.2 Bulk carriers

These ships are constructed to transport substantial amounts of various commodities, including wheat, coal, iron ore, and cement, among others.

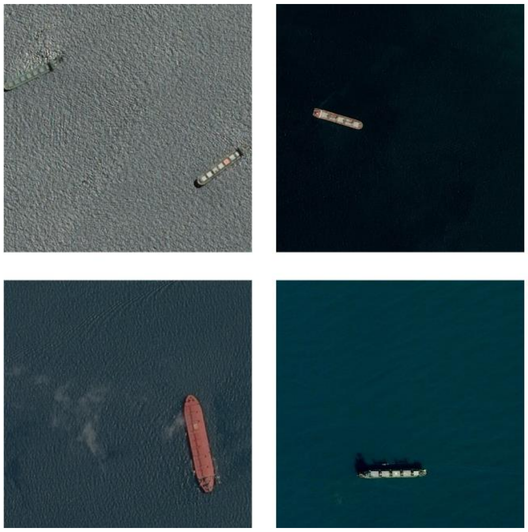


Fig. 5. Bulk Carriers

Notable features of these vessels include open compartments that are vast and may be partitioned into many sections, as well as hull designs that are straightforward and angular.

3.3.3 Container ships

Container ships are specifically designed ships used for the transportation of standardized shipping containers.

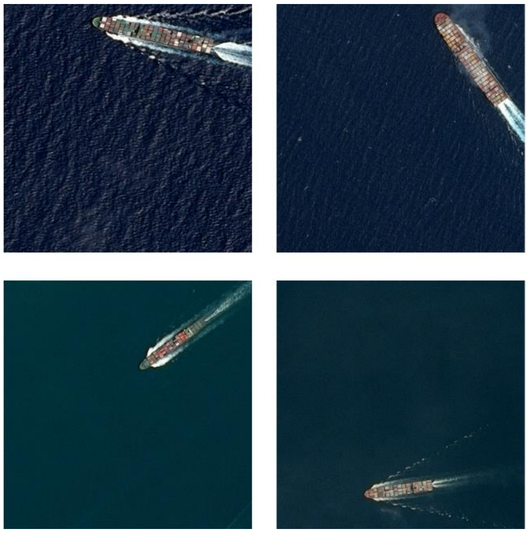


Fig. 6. Container Ships

These containers are stored in spacious cargo holds that are segregated into compartments. The sizes of these vessels differ greatly, with some smaller feeder ships potentially transporting a few hundred containers

© Mehran University of Engineering and Technology 2024

while the largest ultra-container vessels can carry over twenty thousand containers.

3.3.4 Sand carriers

The ships were built with the express goal of delivering aggregates and sand, and they are fully equipped to do so.

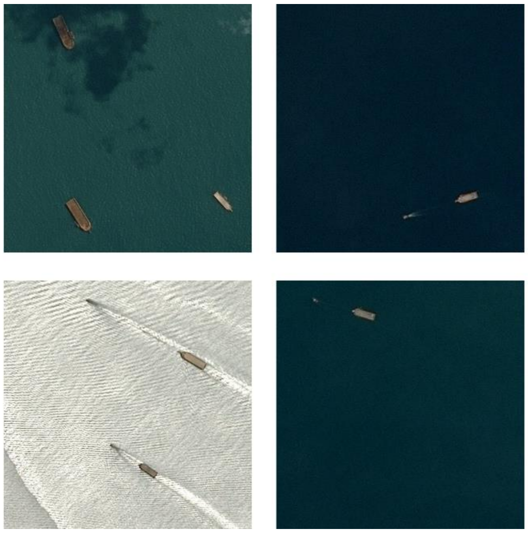


Fig. 7. Sand Carriers

Because they have a shallow draft, they can operate close to the shore. Additionally, the shapes of their hulls are maintained simple.

3.3.5 General ships (Unclassified ships)

General ships are a broad category that encompasses a wide range of vessels that do not fall into any particular classifications or specializations.

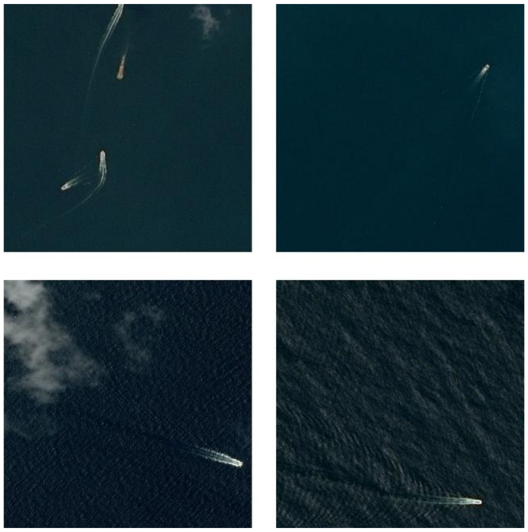


Fig. 8 General Ships (Unclassified Ships)

Boats that are used for fishing, small cruise ships, private vessels, and other types of seagoing vessels might be included in this category.

3.4 YOLOv5 (Object Detection Algorithm)

YOLOv5 is a state-of-the-art object detection model based on a single-stage architecture that predicts the bounding boxes and class probabilities of objects in an image in a single pass. The model is trained using a large dataset of labelled images and employs a novel loss function that combines both localization and classification losses. The mathematical equations for the loss function are:

$$L_{b\otimes} = \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} [\left(\sqrt{w_i} - \sqrt{\hat{w}_i}\right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}}\right)^2] \tag{1}$$

$$L_{obj} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left(C_i - \hat{C}_i\right)^2 \tag{2}$$

$$L_{noobj} = \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{noobj} \left(C_i - \hat{C}_i\right)^2 \tag{3}$$

$$L_{class} = \sum_{i=0}^{S^2} 1_{i}^{obj} \sum_{c\in \text{classes}} \left(pi(c) - \hat{p}(c)\right)^2 \tag{4}$$

Where $L_{b\otimes}$, L_{obj}, L_{noobj}, L_{class} represent the localization, objectness, no-objectness, and classification losses, respectively. The 1_{ij}^{obj} and 1_{ij}^{noobj} terms are indicator functions that determine whether an object is present or not in a particular grid cell, while λ_{coord}, λ_{noobj}, λ_{class} are hyperparameters that control the relative importance of the different loss terms. To enhance the capability of accurately classifying and identifying objects in images, the model has been constructed with these loss functions optimized during the training phase. In Fig. 9. the architecture of YOLOv5 is explained in detail.

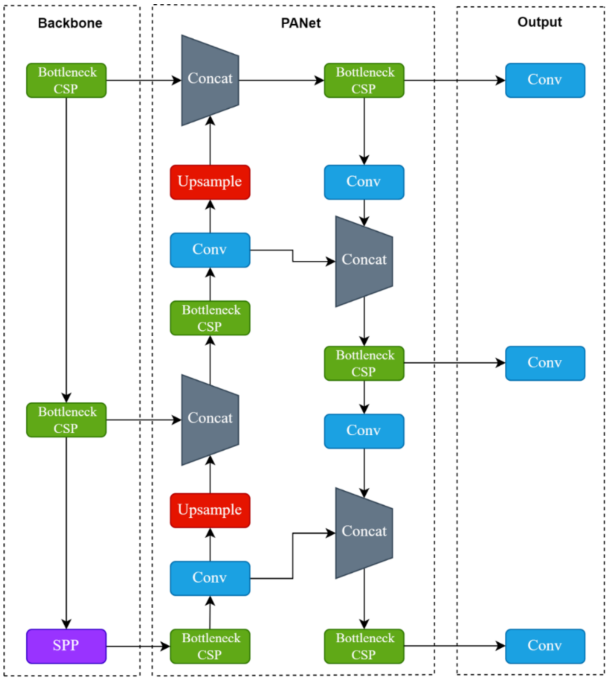


Fig. 9. Architecture of YOLOv5

The pre-configured modes like Yolov5x, Yolov5s, and Yolov5l of the Yolov5 framework are known standouts, and they vary from each other concerning size and complexity. It would be the customer to decide which version most favours him because of his own specific needs for accuracy and performance since the versions tactfully differ regarding amount of layers and functionalities they offer. The newest edition of Yolo, YOLO v5, enables the designing of personal models for follow-up. These models may be built by using a customized dataset from which it is then trained and fine-tuned to fulfil a particular application. Users could leverage this capability to give the algorithms a preference towards detecting given elements and as well achieving efficiency adjusted based on their hardware.

YOLOv5 yields remarkable results in object detection as it integrates advanced deep learning techniques and is optimized to deliver the best possible performance. The model's success is based on a special loss function which is skilfully managing localization, objectness, no-objectness, and classification losses Consequently, the model achieves a high level of precision in recognizing and sorting objects in the images.

3.5 YOLO-C3GTR (Proposed Architecture)

The member YOLOv5x model is altered to develop the YOLO-C3GTR version of the model which includes two C3TR modules at the end of the neck and backbone and a combination of CBAM and C3Ghost modules at different places within the neck. The transition layer enlarges the network as well as increases its capability to obtain the high-level features, and the C3TR module is a boosted version C3 module. The CBAM module acts as a feature-attention mechanism adding on to the accuracy of object recognition by highlighting features of importance.

The C3TR module, out last in the neck and backbone module, is formed by three convolutional layers. The primary purpose of this element is to increase the model's ability to understand intricate relationships and traits found in the input data. This further improves the accuracy level in recognising items in the tasks where such elements are involved.

The implementation of the C3TR units connected to each of the two layers of CBAM and the two C3 modules replaced by C3Ghost was performed. The priority mechanism realized through CBAM layers, strengthens the model's ability to focus on the salient elements in the input data. These layers act as filters that either emphasise or diminish the features depending on their significance. By having CBAM elements, the model can highly focus on the most significant input features that may result in improved precision.

The goal of bringing about the C3Ghost module is the reduction of memory usage by the model and the maintenance of the precision. The C3Ghost module is based on ghost convolution, which is cheaper than the traditional convolutional filter, so we use some of them instead. By doing this, parameters are declined in the model allowing for an easier training process and decreasing the possibility of overfitting. The use of the C3Ghost module is good because it facilitates the implementation of different feature maps with a less computational-intensive representation method. This is because it is possible to somewhat mirror the expensive "Ghost feature maps" and with such techniques, costlier feature maps can be replaced.

The implementation of these modifications does have the benefit of both a step-up of the model's efficiency via a decrease in parameters and computation level and a step-up of the model's capability to understand complex features and patterns in the input data. The addition of the C3TR module to the model serves as a deeper structure, and hence, the chances of accuracy improvement in object detection improve. Another functionality of CBAM layers is that it improves the key element of the attention mechanism thus the model focuses on the most significant part of the inputs. The last benefit of C3Ghost in the training process is the usage of less memory. The result is a simplified training process with a reduced risk of overfitting.

The following is a description of the submodules utilized in the proposed architecture:

3.5.1 C3TR

The C3TR module is an adjusted version of the neck segment of the YOLOv5x model which is formed by the combination of the C3 module and the TR module which is an improved Transformer module. In the C3 module, the traditional convolutional module presents the three convolutional layers with a 3x3 kernel size for the extraction of the image's features.

The TR module likewise TR module is a variation of the self-attention mechanism found in the Transformer model. For this module, the attention heads of the network are accounted for the directing the focus of the system at the diverse regions of the input map is the main function. Among all these heads, each attention headfirst evaluates the importance and relevance of input features and then considers these features as weighted sums on the query vector.

The task of C3TR is to pass the C3 linear output to the

© Mehran University of Engineering and Technology 2024

TR module and here the self-attention is applied using the feature maps. The final output comes about when the Convolutional 3-layer output is combined with the Transposed Convolutional layer output and another Convolutional layer is passed through it.

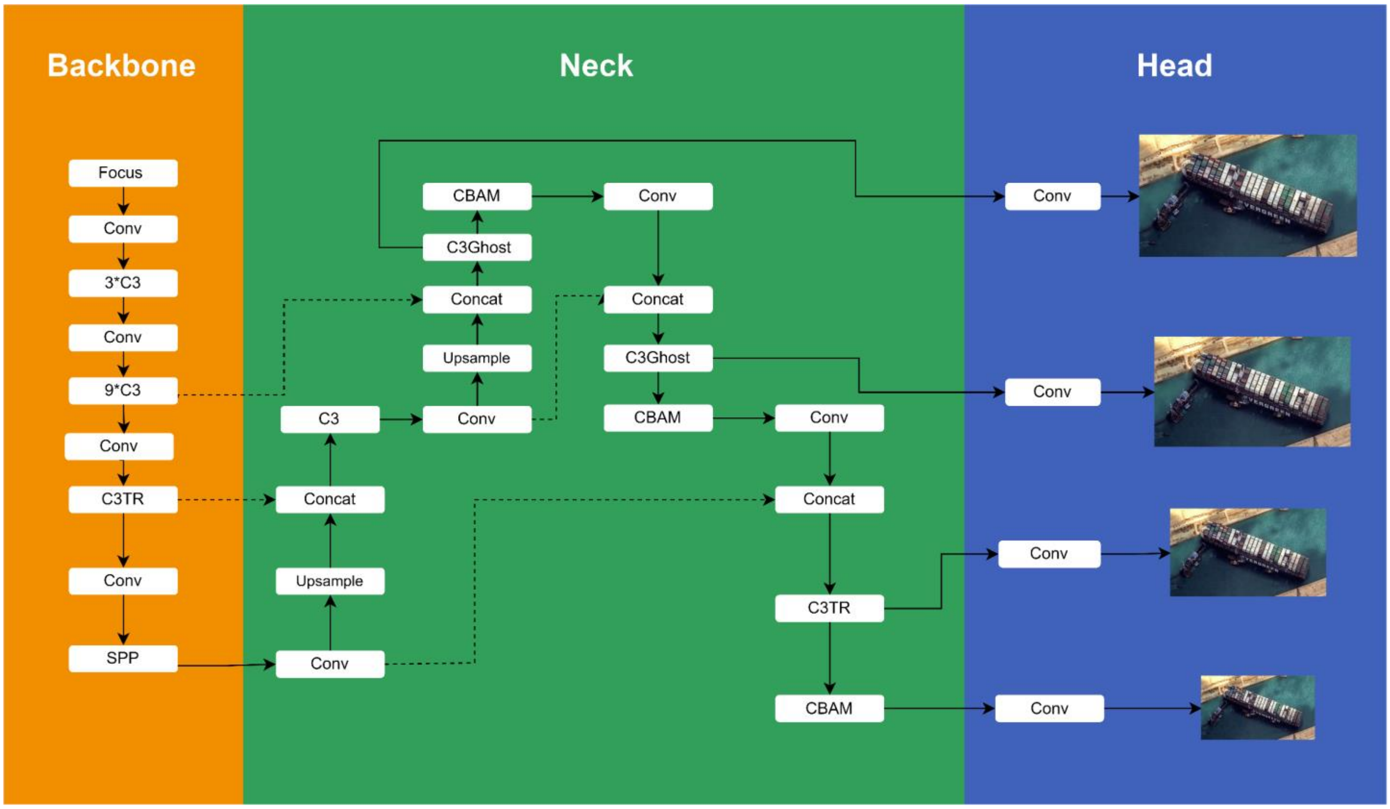


Fig. 10. Schematic of YOLO-C3GTR Proposed Architecture

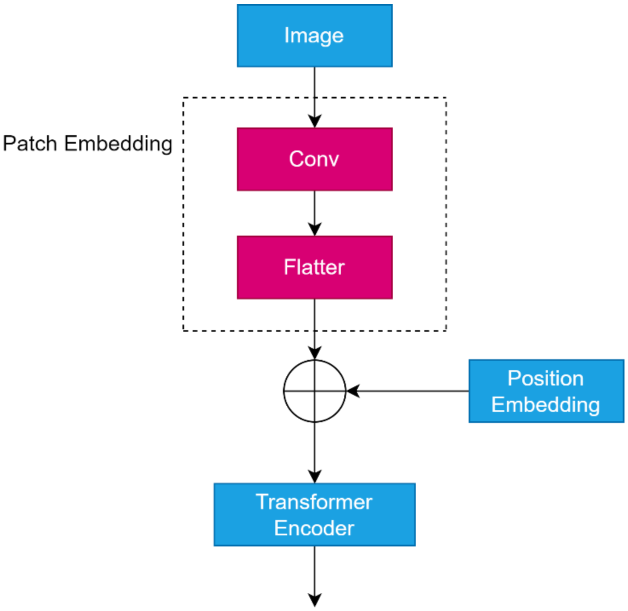


Fig. 11. Architecture of C3TR

The C3TR module boasts several benefits in contrast to the regular C3 module. It provides the capability to the model to focus on particular areas of the feature map which helps to strengthen its ability to perceive complex feature relationships. Firstly, it reduces the amount of parameters in the model allowing for a better generalization and prevention of over-fitting, instead of using separate C3 and TR modules. Not only that, but it has also shown higher performance in many object detection tasks, which creates a functional addition to the YOLOv5x model.

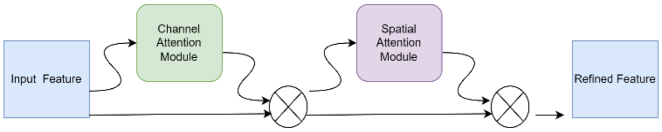


Fig. 12. Architecture of Convolution Block Attention Module

3.5.2 Convolution block attention module (CBAM)

For moving in the direction of overcoming its problems in detecting objects, the custom model YOLO-C3GTR is based on the Convolutional Block Attention Module (CBAM), which is an attention mechanism. This system incorporates an adaptive attention mechanism that enables the network to selectively emphasize essential data elements while getting rid of insignificant ones. These steps are accomplished in two stages. A channel attention block can compute attention maps for each channel thus delivering the importance of a particular feature map per channel. Also, the strategy of computing the spatial attention maps in the feature map endeavours to express the weights for every pixel. The negation of CBAM layers in the YOLO-C3GTR model improves the attention mechanism of the model and makes it concentrate upon the most significant parts of the input datum.

The CBAM component in YOLO-C3GTR becomes the highlight of the model for better object recognition ability of the model by providing it with the capability to recognize main features and patterns in the input data. By fusion, this level of recognition becomes more detailed and precise, and consequently, it aids in the better performance of the networks in object recognition tasks.

3.5.3 C3 ghost

The C3Ghost module is a modified version of the C3 module, which is dedicated to usage in CNNs with the purpose of accomplishing object recognition and categorization. One of the key functionalities of this optimized version is to improve the accuracy of the standard C3 module while simultaneously lowering its computational load.

The C3 module or the convolutional neural networks module serves as the core component for which there are three convolution layers. This layer is not only made to implement a filter set but is also intended to select various characteristic features of the data. However, an exaggeration of the C3 module lies in data processing, especially when working with huge input images or elaborated data.

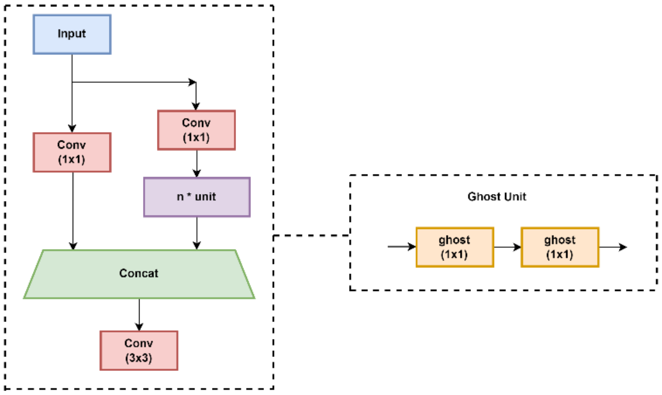


Fig. 13. Architecture of C3Ghost

The problem is resolved by selecting the 'C3Ghost' module option that replaces certain feature maps with cheaper "ghost" feature maps. Opposed to a separate filter bank for every feature map, ghost feature maps come to the user by randomly picking a portion of the filters and then copying them repeatedly. The C3Ghost module may utilize fewer filters that nevertheless preserve the same level of capability in terms of representation if such a method is applied. Therefore, a load of the module's computational burden is dropped.

The CNN may need fewer parameters and calculations if the C3Ghost module is used in place of the conventional C3 module. As a result, there may be less chance of overfitting and an easier-to-manage training procedure.

3.6 Experimental Setup

Our study focuses on ship identification and classification to enhance current methods in this area. To achieve this objective, we meticulously annotated 2000 photos from the Airbus Ship Detection Dataset, categorizing them into five specific ship types: oil tankers, bulk carriers, container ships, sand carriers, and unclassified ships. In order to ensure the efficacy of our deep learning models, we initially divided our dataset into two separate sets - a training set comprising 70% of the images, and a validation set containing the remaining 30%. This labelling process represents an innovative contribution to the field. Subsequently, we trained our labelled images using various deep learning models, such as TPH-Yolo v5, Yolo v5x and our proposed architecture YOLO-C3GTR.

3.6.1 Hyperparameter Tuning

Table 2

List of hyperparameters used to train all models.

Hyperparameters	Values
Initial Learning Rate	0.01
Momentum	0.937
Weight Decay	0.0005
Box Loss Gain	0.05
CLS Loss gain	0.3
IOU Training Threshold	0.2
Image Transition	0.1
Image Scaling	0.9
Image Flip	0.5

The performance of a deep learning model is greatly affected by the hyperparameters utilized during training. In the case of the YOLO-C3GTR model, an initial learning rate of 0.01 has been employed to dictate the rate at which the optimizer adjusts the network's weights during backpropagation. A high learning rate may result in rapid convergence, ultimately yielding subpar outcomes, whereas a low learning rate can prolong the convergence process, thereby increasing the duration of training.

The momentum value of 0.937 assists the optimizer in avoiding local minima and expedites the convergence process. A weight decay of 0.0005 mitigates the risk of overfitting by introducing a penalty term to the loss function that discourages large weights. Additionally, the box loss gain of 0.05 and class loss gain of 0.3 dictate the relative impact of localization and classification losses on the overall loss function. Moreover, the IOU training threshold of 0.2 controls the minimal intersection-over-union (IOU) required between predicted and ground truth boxes for them to be identified as valid detection.

To improve the robustness of the model, we have used data augmentation techniques such as translating the picture by a factor of 0.1, scaling the image by a factor of 0.9, and horizontally flipping the image by a factor of 0.5. These alterations result in improvements to the training pictures, therefore improving the model's capacity to apply knowledge to unfamiliar material. By carefully choosing and fine-tuning these hyperparameters, we have ensured that the YOLO-C3GTR model achieves optimal performance in the assigned job.

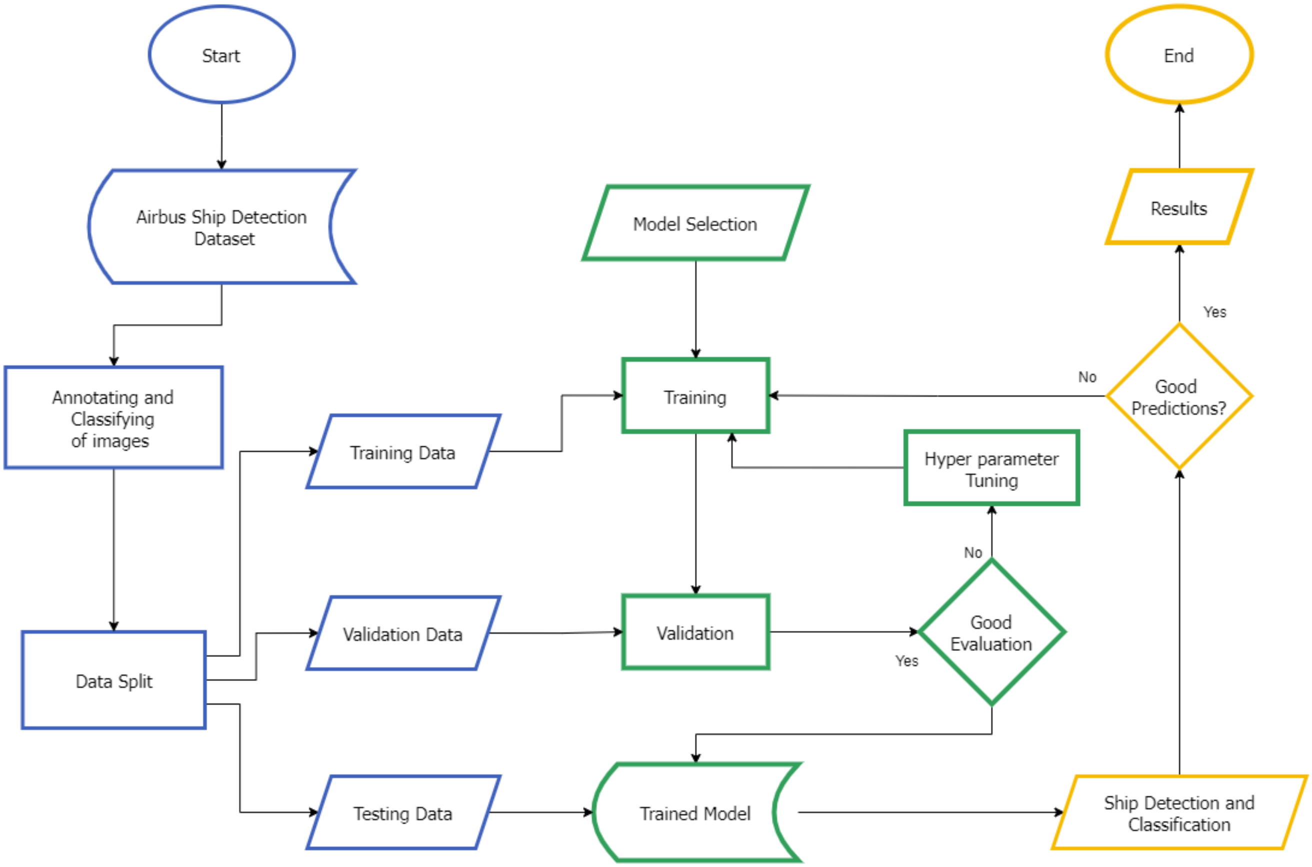


Fig. 14. Flow Diagram of Experimental Process

3.7 Evaluation Metrics

3.7.1 Recall

$$Recall = \frac{TP}{TP+FN} \tag{5}$$

The equation above may be explained as the count of accurately anticipated positive classes out of the total number of positive classes. Maximizing the recall is important.

3.7.2 Precision

$$Precision = \frac{TP}{TP+FP} \tag{6}$$

The equation above can be clarified by examining the ratio of accurately predicted positive classes to the total number of classes anticipated as positive. Maximizing accuracy is essential in this circumstance.

3.7.3 F-measure

$$F - measure = \frac{2 \times Recall \times Precision}{Recall + Precision} \qquad (7)$$

It will not be always easy to contrast two models with a low precision rate and high recall or the other way around. In order then to compare, we have to use F-Score. F-score assists in the measuring of recall and precision all at once. Instead of Arithmetic Mean, it uses Harmonic Mean by exiling the extreme values more.

3.7.4 Mean average precision (mAP)

In object detection, Mean Average Precision (MAP) is a commonly used evaluation metric to assess the performance of a model. MAP measures how well a model can identify and locate objects of interest in an image. It is calculated by first computing the average precision (AP) for each class of object, which represents the precision and recall trade-off for a specific class. AP is then averaged across all classes to obtain the final MAP score. Mathematically, MAP can be expressed as:

$$MAP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision(R_{jk}) \qquad (8)$$

where Q is a set of queries, mj is the number of relevant documents for query j, and Rjk is the set of ranked retrieval results from the top result until you get to document k.

3.7.5 Loss function

The loss function used in an object detection task using YOLO (You Only Look Once) algorithm can be defined as follows:

(a) Localization Loss:

$$\sum_{i=0}^{S^2} \sum_{j=0}^{B} 1ij^{obj} [(\hat{p}ij - pij)^2 + (\hat{b}x, ij - bx, ij)^2 + (\hat{b}y, ij - by, ij)^2 + (\hat{b}w, ij - bw, ij)^2 + (\hat{b}h, ij - bh, ij)^2] \qquad (9)$$

It is the localization loss. It computes the sum of the squared differences between the predicted and true values for the bounding box parameters (*x,y,w,h)* and the confidence score (*p*) of each grid cell *ij* where an object is present $1ij^{obj} = 1$

(b) Classification Loss:

$$\lambda \sum_{i=0}^{S^2} \sum_{j=o}^{B} 1ij^{obj} (\hat{c}ij - cij)^2 \qquad (10)$$

It is the classification loss. It computes the sum of squared differences between the predicted and the true class probabilities (c) of each grid cell *ij* where an object is present. The hyperparameter λ controls the weight of the loss term.

(c) Background Loss:

$$\sum_{i=0}^{S^2} \sum_{j=o}^{B} 1ij^{noobj} (\hat{c}ij - cij)^2 \qquad (11)$$

It is the background loss. It computes the sum of squared differences between the predicted and true class probabilities of each grid cell *ij* where an object is not present $1ij^{obj} = 1$. This statement customizes the model to predict a high confidence score in situations where there is no item present in the cell.

Concisely, this loss function considers the relative significance of localization, classification, and background losses to train the model efficiently in properly detecting and classifying objects in the input picture.

4. Results and Discussion

Our recommended architecture, the YOLO-C3GTR model, obtained an F1 score of 0.79, precision accuracy of 0.81, and recall of 0.78. The results demonstrate the strong accuracy and recall of the model, confirming its effectiveness in correctly identifying and categorizing ships in the dataset with low rates of false positives and false negatives. The F1 score of 0.79 demonstrates that the YOLO-C3GTR model effectively achieves a harmonious equilibrium between recall and accuracy, rendering it a highly suitable option for tasks involving the identification and classification of ships.

Table 3

Ship Detection comparison results on Airbus Dataset

Architecture	Precision	Recall	F1 Score	$mAP_{0.5}$
TPH-YOLOv5	0.68	0.11	0.18	0.363
YOLOv5x	0.79	0.77	0.77	0.805
YOLOv5l	0.80	0.78	0.78	0.816
YOLO-C3GTR	0.81	0.78	0.79	0.835

The YOLO-C3GTR model (see Fig. 15) registers the highest mean average precision (mAP) value of 0.835 among all the models. The precise ship recognizing, and classification algorithms of this model can be confirmed by its high mean average precision (mAP) score factor. The outstanding mean average precision (mAP) score of the YOLO-C3GTR model is an indication of its capability to classify and label the ships accurately in the dataset to minimize the risk of false positives and negative detection.

It is shown in the image comparison that the mAP for the YOLO-C3GTR model is 0.835, while for YOLOv5x the mAP is 0.805, which makes the model of YOLO-C3GTR better than the YOLOv5x model by approximately 3.5 %. On the other hand, the TPHYOLO v5 model has its worst mAP value amongst the compared models and it is at 0.363. The above findings reveal that the YOLO-C3GTR model is the best model for detecting and categorising the ship class in this dataset.

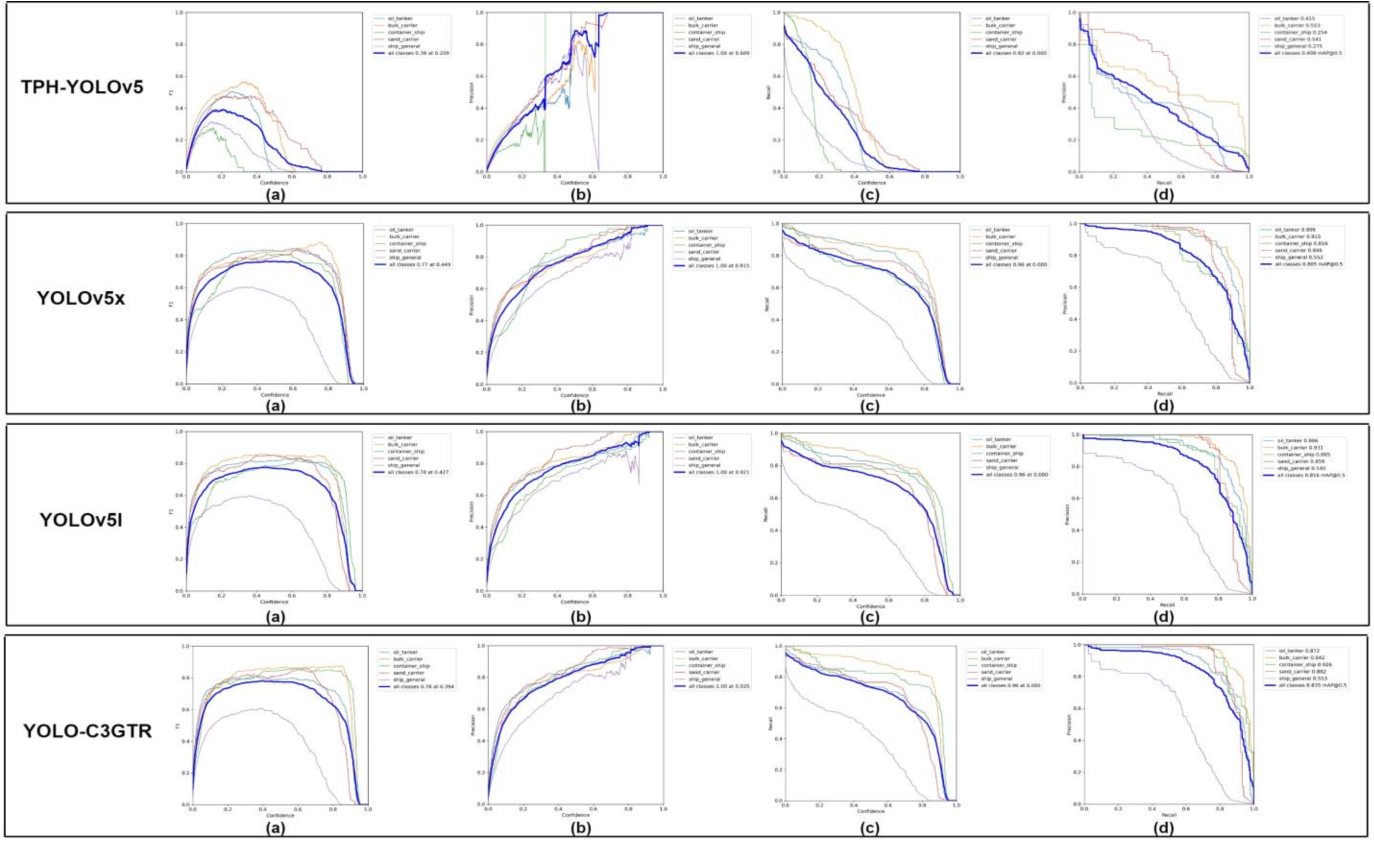


Fig. 15. Results Comparison of Different Object Detection Architectures for Airbus Ship Detection Dataset. (a) F1 Score, (b) Precision, (c) Recall, (d) mAP

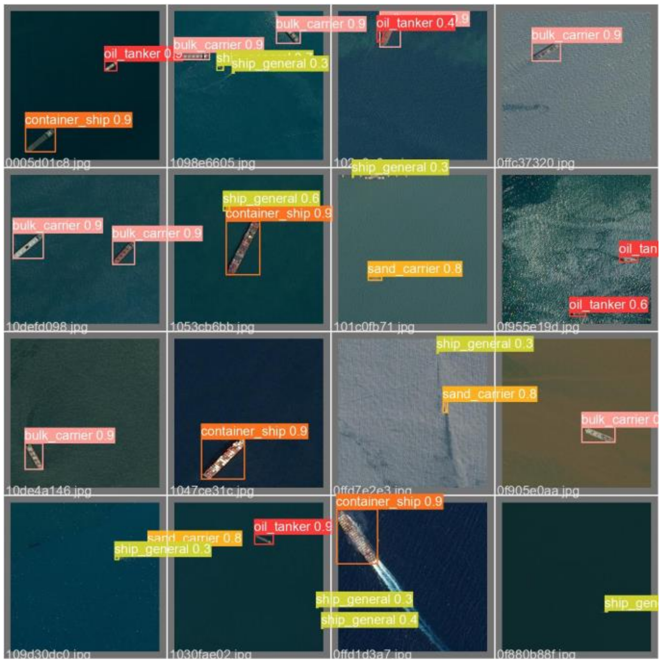
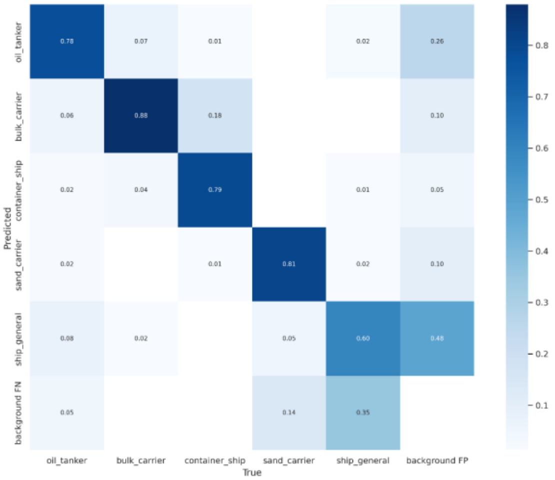


Fig. 16. Results Of Predicted Ship Classes Through The YOLO-C3GTR Model



Predicted \ True	oil_tanker	bulk_carrier	container_ship	sand_carrier	ship_general	background FP
oil_tanker	0.78	0.07	0.01		0.02	0.26
bulk_carrier	0.06	0.88	0.18			0.10
container_ship	0.02	0.04	0.79		0.01	0.05
sand_carrier	0.02		0.01	0.81	0.02	0.10
ship_general	0.08	0.02		0.05	0.60	0.48
background FN	0.05			0.14	0.35	

Fig. 17. Confusion Matrix of YOLO-C3GTR

The metrics of precision, recall, and F1 score are commonly utilized for binary classification tasks, and they do not have a direct correlation with mAP. Nonetheless, in object detection assignments, precision and recall are determined individually for each class, and subsequently, the Average Precision (AP) is

© Mehran University of Engineering and Technology 2024

computed for each class. The map represents the average of AP values across all classes.

5. Conclusion

Our study aimed to enhance the identification and categorization of ships through the utilization of advanced learning models. We manually labelled 2000 images from the Airbus Ship Detection Dataset and categorized them into five distinct ship types. Despite training our labelled dataset with various deep-learning models, we found that they did not meet our desired standards for accuracy and speed. We implemented a novel architecture called YOLO-C3GTR, which integrates changes including C3TR modules after the backbone and neck, CBAM and C3Ghost modules at various locations in the neck, and the substitution of certain C3 modules with C3Ghost.

The YOLO-C3GTR model demonstrated superior performance in both accuracy and speed compared to the other models assessed. The regarded model has proven to be superlative as it accomplished an accuracy rate of 84% when its resilience was tested with an independent dataset. Apart from that, we also performed calculations for accuracy, recall, F1-score, and mAP which resulted in values of 0.81, 0.78, 0.79, and 0.835, respectively. Additionally, it showed more rapid inference capabilities than the other model, making it possible to execute for real-time ship classification and identification.

The study offers multiple ways of sophisticating the ship's detection and categories considering the employment of deep learning models. Survey the different components and framework selection, including EfficientDet and RetinaNet, within our framework, could be the way forward. These models have been shown to work well for other cognate object identification problems, and hence, seem to be a suitable candidate for adapting to the ship recognition and classification task. Subsequently, this creates the possibility of hyperparameters investigation, and the idea is to find the best values that can lead to better accuracy and speed. Besides, looking at the enlargement of the annotated dataset as a potential for improving our model performance, namely, to correctly classify less typical designs of ships is a very promising idea.

6. References

[1] M. Waseem Sabir, M. Farhan, N. S. Almalki, M. M. Alnfiai, and G. A. Sampedro, “FibroVit—vision transformer-based framework for detection and classification of pulmonary fibrosis from chest ct images,” Front Med (Lausanne), vol. 10, 2023, doi: 10.3389/fmed.2023.1282200.

[2] U. Javaid, R. M. Usman, and A. Javaid, “Investigating the energy production through sustainable sources by incorporating multifarious machine learning methodologies,” 3rd IEEE International Conference on Artificial Intelligence, ICAI 2023, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 233–237. doi: 10.1109/ICAI58407.2023.10136677.

[3] M. Doumpos, C. Zopounidis, D. Gounopoulos, E. Platanakis, and W. Zhang, “Operational research and artificial intelligence methods in banking,” European Journal of Operational Research, vol. 306, no. 1. Elsevier B.V., pp. 1–16, Apr. 01, 2023. doi: 10.1016/j.ejor.2022.04.027.

[4] A. A. Minhas, S. Jabbar, M. Farhan, and M. Najam ul Islam, “A smart analysis of driver fatigue and drowsiness detection using convolutional neural networks,” Multimed Tools Appl, vol. 81, no. 19, pp. 26969–26986, Aug. 2022, doi: 10.1007/s11042-022-13193-4.

[5] S. A. Haider, M. Sajid, H. Sajid, E. Uddin, and Y. Ayaz, “Deep learning and statistical methods for short- and long-term solar irradiance forecasting for Islamabad,” Renew Energy, vol. 198, pp. 51–60, Oct. 2022, doi: 10.1016/j.renene.2022.07.136.

[6] S. A. Haider, M. Sajid, and S. Iqbal, “Forecasting hydrogen production potential in Islamabad from solar energy using water electrolysis,” Int J Hydrogen Energy, vol. 46, no. 2, pp. 1671–1681, Jan. 2021, doi: 10.1016/j.ijhydene.2020.10.059.

[7] A. Javaid et al., “Forecasting hydrogen production from wind energy in a suburban environment using machine learning,” Energies (Basel), vol. 15, no. 23, Dec. 2022, doi: 10.3390/en15238901.

[8] A. Javaid, M. F. Faiz, and R. M. Usman, “Shifting energy horizons: utilizing ai-driven energy forecasting for hydrogen production in urban environments,” 2023 International Conference on Digital Futures and Transformative Technologies, ICoDT2 2023, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ICoDT259378.2023.10325817.

[9] A. Javaid, M. Sajid, E. Uddin, A. Waqas, and Y. Ayaz, “Sustainable urban energy solutions: forecasting energy production for hybrid solar-wind systems,” Energy Convers Manag, vol. 302, p. 118120, Feb. 2024, doi: 10.1016/j.enconman.2024.118120.

[10] A. P. Saputra, “Waste object detection and classification using deep learning algorithm: YOLOv4 and YOLOv4-tiny,” 2021.

[11] X. Wang, T. Kong, C. Shen, Y. Jiang, and L. Li, “SOLO: Segmenting Objects by Locations,” Dec. 2019, doi: doi.org/10.48550/arXiv.1912.04488.

[12] W. Liu et al., “SSD: Single shot multibox detector,” 2016, pp. 21–37. doi: 10.1007/978-3-319-46448-0_2.

[13] Y. Zhu, C. Zhao, J. Wang, X. Zhao, Y. Wu, and H. Lu, “CoupleNet: Coupling global structure with local parts for object detection,” 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, Oct. 2017, pp. 4146–4154. doi: 10.1109/ICCV.2017.444.

[14] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, “Single-shot refinement neural network for object detection,” Nov. 2017, [Online]. Available: http://arxiv.org/abs/1711.06897

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: unified, real-time object detection,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,”2014 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Jun. 2014, pp. 580–587. doi: 10.1109/CVPR.2014.81.

[17] R. Girshick, “Fast R-CNN,” 2015 IEEE International Conference on Computer Vision (ICCV), IEEE, Dec. 2015, pp. 1440–1448. doi: 10.1109/ICCV.2015.169.

[18] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: towards real-time object detection with region proposal networks,” IEEE Trans Pattern Anal Mach Intell, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[19] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, Oct. 2017, pp. 2980–2988. doi: 10.1109/ICCV.2017.322.

[20] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Jun. 2015, pp. 3431–3440. doi: 10.1109/CVPR.2015.7298965.

[21] Q. Wang, J. Gao, and X. Li, “Weakly supervised adversarial domain adaptation for semantic segmentation in urban scenes,” Apr. 2019, doi: 10.1109/TIP.2019.2910667.

[22] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” Dec. 2015.

[23] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Jul. 2017, pp. 2261–2269. doi: 10.1109/CVPR.2017.243.

[24] Y. Hu, X. Li, N. Zhou, L. Yang, L. Peng, and S. Xiao, “A sample update-based convolutional neural network framework for object detection in large-area remote sensing images,” IEEE Geoscience and Remote Sensing Letters, vol. 16, no. 6, pp. 947–951, Jun. 2019, doi: 10.1109/LGRS.2018.2889247.

[25] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal speed and accuracy of object detection,” Apr. 2020.

[26] A. Van Etten, “You only look twice: rapid multi-scale object detection in satellite imagery,” May 2018.

[27] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Jul. 2017, pp. 6517–6525. doi: 10.1109/CVPR.2017.690.

[28] J. Tang, C. Deng, G.-B. Huang, and B. Zhao, “Compressed-domain ship detection on spaceborne optical image using deep neural network and extreme learning machine,” IEEE Transactions on Geoscience and Remote Sensing, vol. 53, no. 3, pp. 1174–1185, Mar. 2015, doi: 10.1109/TGRS.2014.2335751.

[29] Z. Zou and Z. Shi, “Ship detection in spaceborne optical image with svd networks,” IEEE Transactions on Geoscience and Remote Sensing, vol. 54, no. 10, pp. 5832–5845, Oct. 2016, doi: 10.1109/TGRS.2016.2572736.

[30] Q. Li, L. Mou, Q. Liu, Y. Wang, and X. X. Zhu, “HSF-Net: Multiscale deep feature embedding for ship detection in optical remote sensing imagery,” IEEE Transactions on Geoscience and Remote Sensing, vol. 56, no. 12, pp. 7147–7161, Dec. 2018, doi: 10.1109/TGRS.2018.2848901.

[31] F. Yang, Q. Xu, and B. Li, “Ship detection from optical satellite images based on saliency segmentation and structure-LBP feature,” IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 5, pp. 602–606, May 2017, doi: 10.1109/LGRS.2017.2664118.

[32] G. Cheng, P. Zhou, and J. Han, “Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images,” IEEE Transactions on Geoscience and Remote Sensing, vol. 54, no. 12, pp. 7405–7415, Dec. 2016, doi: 10.1109/TGRS.2016.2601622.

[33] R. W. Liu, W. Yuan, X. Chen, and Y. Lu, “An enhanced CNN-enabled learning method for promoting ship detection in maritime surveillance system,” Ocean Engineering, vol. 235, p. 109435, Sep. 2021, doi: 10.1016/j.oceaneng.2021.109435.

[34] Y. Zhao, L. Zhao, B. Xiong, and G. Kuang, “Attention receptive pyramid network for ship detection in sar images,” IEEE J Sel Top Appl Earth Obs Remote Sens, vol. 13, pp. 2738–2756, 2020, doi: 10.1109/JSTARS.2020.2997081.