# CNN-based Boat Detection for Environmental Protection Area Monitoring ⋆

**Vitor G. Santos** * **Diego S. Pereira** * **Luis B. P. Nascimento** ** **Pablo J. Alsina** **

* *Federal Institute of Rio Grande do Norte, Parnamirim, RN, (e-mail: vitor.gaboardi@ifrn.edu.br, diego.pereira@ifrn.edu.br).*
** *Department of Computer Engineering and Automation, Federal University of Rio Grande do Norte, Natal, RN, (e-mail: lbruno@ufrn.edu.br, pablo@dca.ufrn.br).*

**Abstract:** Boats and ships have always been used throughout history as one of the main types of transportation. In recent years, due to the fast evolution of deep learning techniques and online datasets available, convolutional neural networks (CNN) have been widely used for ship and boat detection applications, such as surveillance of marine resources, helping in maritime rescue, monitoring illegal marine activities, among others. In this paper, we present a robust and efficient CNN-based on state-of-the-art YOLO model to perform boat and other water vehicles detection. The training dataset was built considering boats of different sizes, located on the coast and sea and taken with drones and satellites. We also applied data augmentation techniques such as flipping, cropping and changing brightness to increase the number of samples and improve the model robustness. A case study is presented considering a multi Unmanned Aerial Vehicles (UAV) to detect boats in a Coral Reefs Environmental Protection Area (APARC), where human activity is limited. We evaluated the developed system considering a testing dataset with images of the case study, achieving a recognition rate of 87,2% and a mean average precision of 97,23%.

*Keywords:* Boat, Object Detection, Convolutional Neural Network, YOLO, Unmanned Aerial Vehicles, Environmental Protection Area.

## 1. INTRODUCTION

Preserving and protecting the environment is essential nowadays. Climate change, pollution, and overexploitation, among other threats, are severely harming natural resources, biodiversity, ecosystems, and human health (Fascista, 2022). In this way, government initiatives have sought to mitigate these impacts through public policies and incentives for actions related to nature preservation and sustainable use of natural resources. Among them is the 9.985 Brazilian law that defined the Nature Conservation Units (UCs).

The UCs are territorial spaces created for conservation purposes, where their environmental resources are legally established by the government. Environmental Protection Area is a UC category with an extensive area and a certain degree of human occupation that has the goal of protecting the biological environment, disciplining the occupation process and ensuring the sustainability of the natural resources use.

The Coral Reefs Environmental Protection Area (Área de Proteção Ambiental Recifes de Corais - APARC, in portuguese) is a UC located the in the coastal strip from the cities of Maxaranguape, Rio do Fogo and Touros on the Brazilian state of Rio Grande do Norte. This region contains several coral reefs and a diverse marine life, making it an unique place for recreational and tourist diving, artisanal fishing and scientific research. Figure 1 illustrates the APARC.

Figure 1. Coral Reefs Environmental Protection Area (APARC)

The APARC is divided in four zones: Fishing, Tourism, Recreated Diving and Full Protection (Lopes et al., 2014). The first three zones allow controlled local and tourist occupation restricting the number of travelers and stablishing conducting rules that must be followed. However, the Full Protection zone is formed by areas with

⋆ This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

rich and fragile ecosystems, which justifies their protection in relation to any exploitation of natural resources. In this zone, only scientific research, environmental education activities and environmental monitoring are allowed.

Therefore, monitoring actions (mainly in the Full Protection zone) is important to ensure the APARC conservation. Currently, there is a monitoring program, but the extensive area and the limited resources difficults the daily execution. In this sense, an interesting solution to improve monitoring is to use Unmanned Aerial Vehicles (UAVs) with cameras to automatically detect human activities. Thanks to their aerial abilities, UAVs can reach remote and poorly accessible areas and perform monitoring activities at different altitudes while ensuring high sampling rates.

However, using a single UAV may not be enough to complete a mission. In these cases, a viable alternative is to use a fleet of UAVs, in which they must be able to exchange information with each other and act cooperatively to achieve a common goal. The literature calls this solution a system of multiple UAVs or multi-UAV (Fu et al., 2019).

According to Yanmaz et al. (2018), multi-UAV systems present some considerable advantages when compared to single UAV solutions. Among them, we can highlight: greater area coverage and reliability (reception of distinct observations from a particular area of interest); redundancy and fault tolerance (the collective nature inserts fault tolerance in isolated points); availability of resources (can provide an increase in data storage capacity and the installation of specific sensors for each aircraft); and scalability (the inclusion or removal of aircraft allows working on problems of similar nature, but with different proportions).

In this context, our previous work (Santos et al., 2019) presented a multi-UAV architecture for monitoring human activity in the APARC. The main contributions were establishing an UAV network to send images, a flight formation strategy to capture images using UAVs, and testing a pre-trained SSD Neural Network to detect boats in forbidden regions. However, the SSD network did not performed well considering the testing dataset.

In this paper, our main contribution is the development of a convolutional neural network (CNN) to detect boat, ships and other water vehicles located at coast and sea. This network will be applied for monitoring human activies in an Environmental Protection Area using a multi-UAV architecture. Another contribution is the creation of a new public test dataset with fully anotated aerial images of boats in the APARC.

The remainder of this paper is organized as follows: Section 2 presents related works to boat and ship detection. Section 3 describes the dataset used to train and evaluate the network. Section 4 describes the proposed CNN architecture to detect boats. Section 5 reports and discuss results of experiments. Finally, section 6 presents the final considerations.

## 2. RELATED WORKS

In recent years, many studies related to ships and boats detection have been developed with applications in different fields. For example, it is possible to apply this technology in surveillance of marine resources, helping in maritime rescue, monitoring illegal marine activities, securing traffic in ports, among others (Bo et al., 2021).

Most of early works uses satellite images to train and evaluate their system. In Yang et al. (2013), an automatic ship detection using high-resolution optical satellite image based on image processing and sea surface analysis is proposed. They initially analyze texture and intensity information of pixels to detect objects over the ocean. Next, a linear selection function is applied to acquire ship candidates, and then features related to perimeter and length-width ratio are used to assure that the object represents a ship.

Zou and Shi (2016) present a novel ship detection method called SVDNet, created based on convolutional neural networks (CNN) and singular value decompensation algorithm. To supress undesired background and detect ship candidates, the authors use three convolutional layers and three nonlinear mapping layers. Then, each ship candidate is tested using feature pooling operation and a linear SVM classifier. The framework is trained using spaceborne optical images and was evaluated considering ships on the coast and ocean.

Other works also focus on detecting ships considering satellite images but using different neural network architectures. Nie et al. (2017) use Single Shot MultiBox Detector (SSD) with transfer learning while Wang et al. (2019) employs a RetinaNet architecture using multi-resolution images.

In Chen et al. (2020b), a novel ship detection architecture is developed considering non-aerial images acquired by the authors and online. They used Generative Adversarial Networks (GAN) and YOLOv2 and the main goal of this work is to detect small ships such as bamboo rafts and fishing boats in the river or near-shore sea, achieving an accuracy of up to 97.2%.

Similarly, Li et al. (2021) acquired ship images on coastal and river routes using surveillance videos to train the network, i.e., non-aerial images. The authors present an Enhanced YOLOv3 tiny network to detect six different types of boats, which provides a better trade-off between accuracy and processing time.

Lodeiro-Santiago et al. (2019) provide a solution to detect small boats used for irregular immigration. In this work, the images are captured using a smartphone attached to an UAV and the entire frame is sent to a remote cloud server, where a CNN is applied to detect ships, pateras (or cayucos), and people.

## 3. DATASET

In the literature, there are many ship and boat datasets available online that focus mainly on aerial (MakeML, 2020; Gąsienica-Józkowy et al., 2021), satellite (Antonio-Javier Gallego, 2018; Chen et al., 2020a) or both (Zhang et al., 2020) images. In this paper, the dataset used to train and validate the neural network contains approximately 7,900 images provided by three datasets (Antonio-Javier Gallego, 2018; MakeML, 2020; Gąsienica-Józkowy et al., 2021).

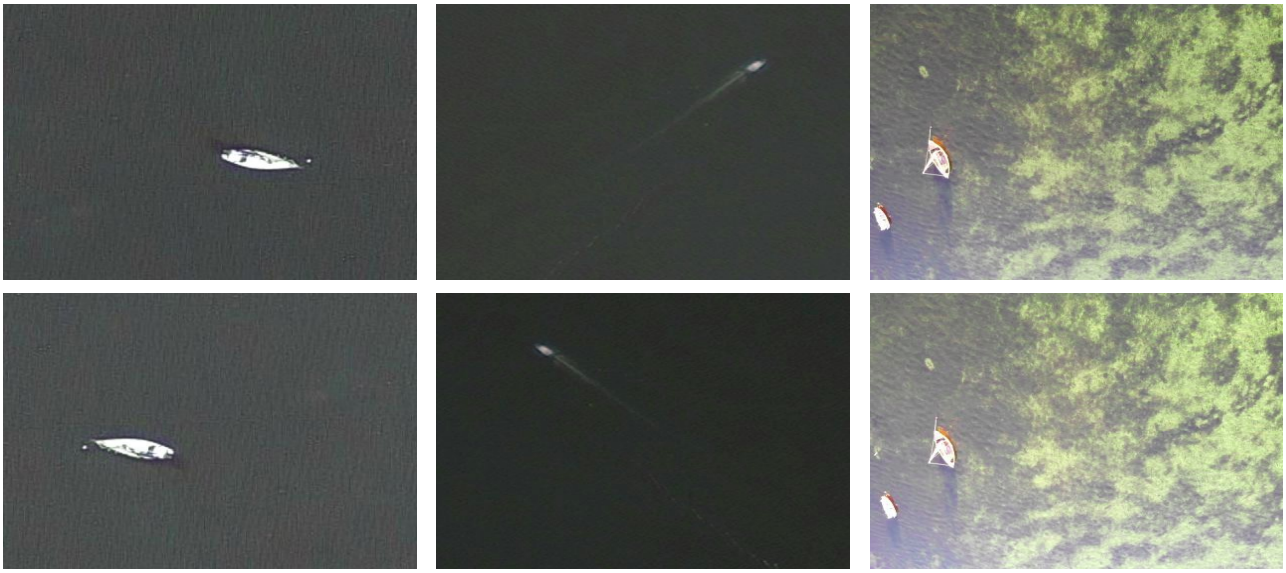

Figure 2. Sample images with data augmentation from training datasets: MakeML (2020) (left), Antonio-Javier Gallego (2018) (middle) and Gąsienica-Józkowy et al. (2021) (right)

These datasets can be summarized as follows: (1) MakeML (2020) has 621 aerial images of ships and boats on the coast and sea; (2) Antonio-Javier Gallego (2018) has 7,389 satellite images labeled in seven classes: land, coast, sea, ship, multi, coast-ship, and detail. The distance between targets and the acquisition satellite is changed to obtain captures at different altitudes; and (3) Gąsienica-Józkowy et al. (2021) has 3,647 images taken from video clips captured by various drone-mounted cameras and labeled in six classes: human, wind/sup-board, boat, buoy, sailboat, and kayak. Images from all datasets are labeled and have bounding boxes of the objects.

In this paper, we focus on detecting boats and other similar vehicles on the sea. However, some of the mentioned datasets have images and classes that does not fit this purpose, which required us to preprocess and filter out some images and redefine some classes. Thus, for dataset (2), we selected only images with the classes ship (1,027 images of singles ships on the sea) and multi (304 images with more than one ship), defining both of them as only one class. Similarly, for dataset (3), we selected images with the classes wind/sup-board, boat, sailboat, and kayak, considering them as only one class, and disregarded the bounding boxes of the classes human and buoy.

The images of these datasets have boats and ships of distinct sizes, located on the ocean and coast, taken with UAVs and satellites at different angles. This diverse image configuration can help improve network generalization, detecting boats from different perspectives. Together, after preprocessing, there are 3,219 images which will be divided in 70% for training and 30% for validation.

We also performed data augmentation to improve the model robustness by introducing new samples to the training dataset. Each training instance has three outputs by executing the following operations: flip each image horizontally or vertically; randomly crop the image by zooming up to 40%; and change brightness between -20% and +20%. These procedures work well with our dataset because flipping will cause no harm since most images were taken directly over the object; zooming up is equivalent to lower the altitude of the drone that took the image and changing the brightness simulates different sunlight levels. The training dataset increased to 6,756 images after data augmentation. Figure 2 shows some examples of each dataset alongside a data augmentation technique.

To test the performance of the neural network, we built our own test dataset from 17 high resolution images (4000x3000 pixels) taken in APARC using a DJI Phantom 3 Standard Quadcopter Drone provided by the Institute of Sustainable Development and Environment of Rio Grande do Norte (IDEMA). Some images show the same boats but from a different angles and lightning conditions. We decided to split them in 98 smaller images with a 400x300 resolution, where each image has at least one boat. In total, there are 219 boats in this dataset randomly positioned within the images. Figure 3 shows a sample image of the testing dataset and Figure 4 resumes the dataset explained in this section.

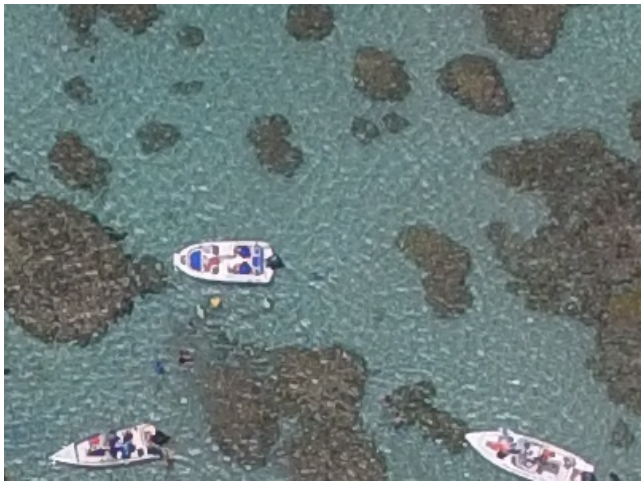

Figure 3. Sample of testing dataset.

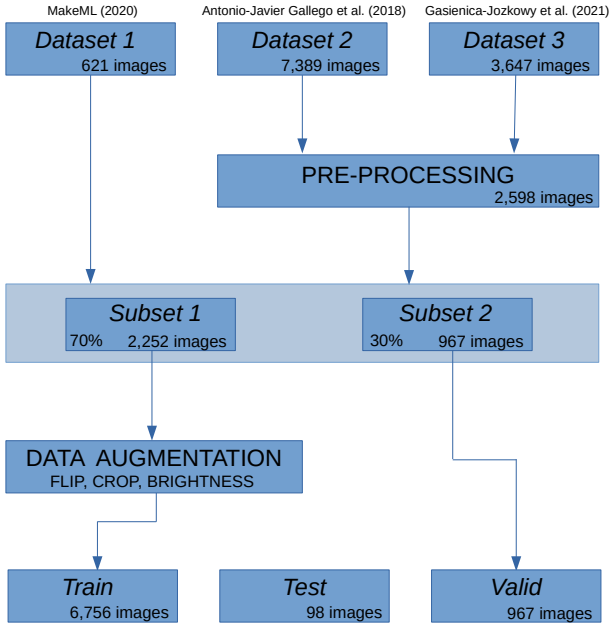


Figure 4. Dataset creation.

The test dataset was manually labeled by the authors and is publicly available for research purposes at Mendeley Data over the name of this paper's title. Every image has a text file in which each line corresponds to one object's position (in our case, a boat) in the image. This position has four coordinates shown in a sequence that represents the x- and y-axis position of the upper-left vertice of the object's bounding box, the width and the height of the object. All coordinates are expressed in pixels and we also resized the images to 416x416 to match the network input.

## 4. CNN ARCHITECTURE

Two main approaches can be used for object detection considering CNN. The first one is a two-stage object detection, used on R-CNN (Girshick et al., 2014) and FPN (Lin et al., 2017). It consists of two networks: the region proposal network, which predicts bounding boxes of the classes; and the object detection network, which classifies these candidate regions and redefines the object localization. In contrast, the second approach is a one-stage object detection and used on YOLO (Bochkovskiy et al., 2020) and SSD (Liu et al., 2016), predicting both bounding boxes with associated class probabilities in a single step. Two-stage object detection approaches have higher accuracy, with the drawback of higher processing time.

The neural network developed in this paper will be deployed on an edge device attached to a UAV. Therefore, due to the edge device processing and storage limitations, we need to balance a lightweight network architecture with satisfactory accuracy. In this context, we used YOLOv4-tiny, a lighter version of YOLOv4 developed for edge and lower-power devices. YOLOv4-tiny has been used for many real-time object detections applications, such as pothole recognition (Silva et al., 2020), complex road scenarios (Zhu et al., 2021), and aircraft detection (Hou et al., 2021), providing a satisfactory trade-off between accuracy and time processing.

To use YOLO-based models, we have to change the number of filters ($F$) in the last convolutional layer based on the number of classes ($C$), anchor boxes ($A$), and coordinates of each bounding box, as pointed out by the equation below.

$$F = (C + 5) \times A \tag{1}$$

For this paper, we only have one class that represents boats and other similar water vehicles, and we will consider 3 anchor boxes in each last convolutional layer. These values result on 18 filters before each YOLO instance. Table 1 resumes the YOLOv4-tiny architecture used in this work, where layers 29 and 36 have 18 filters. Also, the input image is resized to 416x416 pixels.

Table 1. YOLOv4-tiny architecture.

| Layer | Type | Filters | Size/Stride | Input |
|---|---|---|---|---|
| 0 | Conv. | 32 | 3x3/2 | 416x416x3 |
| 1 | Conv. | 64 | 3x3/2 | 208x208x32 |
| 2 | Conv. | 64 | 3x3/1 | 104x104x64 |
| 3 | Route 2 | | | |
| 4 | Conv. | 32 | 3x3/1 | 104x104x32 |
| 5 | Conv. | 32 | 3x3/1 | 104x104x32 |
| 6 | Route 5 4 | | | |
| 7 | Conv. | 64 | 1x1/1 | 104x104x64 |
| 8 | Route 2 7 | | | |
| 9 | Max Pool | | 2x2/2 | 104x104x128 |
| 10 | Conv. | 128 | 3x3/1 | 52x52x128 |
| 11 | Route 10 | | | |
| 12 | Conv. | 64 | 3x3/1 | 52x52x64 |
| 13 | Conv. | 64 | 3x3/1 | 52x52x64 |
| 14 | Route 13 12 | | | |
| 15 | Conv. | 128 | 1x1/1 | 52x52x128 |
| 16 | Route 10 15 | | | |
| 17 | Max Pool | | 1x1/1 | 52x52x256 |
| 18 | Conv. | 256 | 3x3/1 | 26x26x256 |
| 19 | Route 18 | | | |
| 20 | Conv. | 128 | 3x3/1 | 26x26x128 |
| 21 | Conv. | 128 | 3x3/1 | 26x26x128 |
| 22 | Route 21 20 | | | |
| 23 | Conv. | 256 | 1x1/1 | 26x26x256 |
| 24 | Route 18 23 | | | |
| 25 | Max Pool | | 2x2/2 | 26x26x512 |
| 26 | Conv. | 512 | 3x3/1 | 13x13x512 |
| 27 | Conv. | 256 | 1x1/1 | 13x13x512 |
| 28 | Conv. | 512 | 3x3/1 | 13x13x256 |
| 29 | Conv. | 18 | 1x1/1 | 13x13x512 |
| 30 | YOLO | | | |
| 31 | Route 27 | | | |
| 32 | Conv. | 128 | 1x1/1 | 13x13x256 |
| 33 | Up Sample | | 2x | 13x13x128 |
| 34 | Route 33 23 | | | |
| 35 | Conv. | 256 | 3x3/1 | 26x26x384 |
| 36 | Conv. | 18 | 1x1/1 | 26x26x256 |
| 37 | YOLO | | | |

The training was performed considering 7,000 iterations (max batches) and learning rate = $[2,61 * 10^{-3}$,$2,61 * 10^{-4}$,$2,61 * 10^{-5}]$ with steps at 5,600 and 6,300 iterations.

## 5. RESULTS

In this section, we will present and discuss the experiments to evaluate our system considering the test dataset described in Section 3. To perform the tests, we used Google Colab with a Tesla K80 GPU and 12 GB of memory using the Darknet framework (Redmon, 2013–2022).

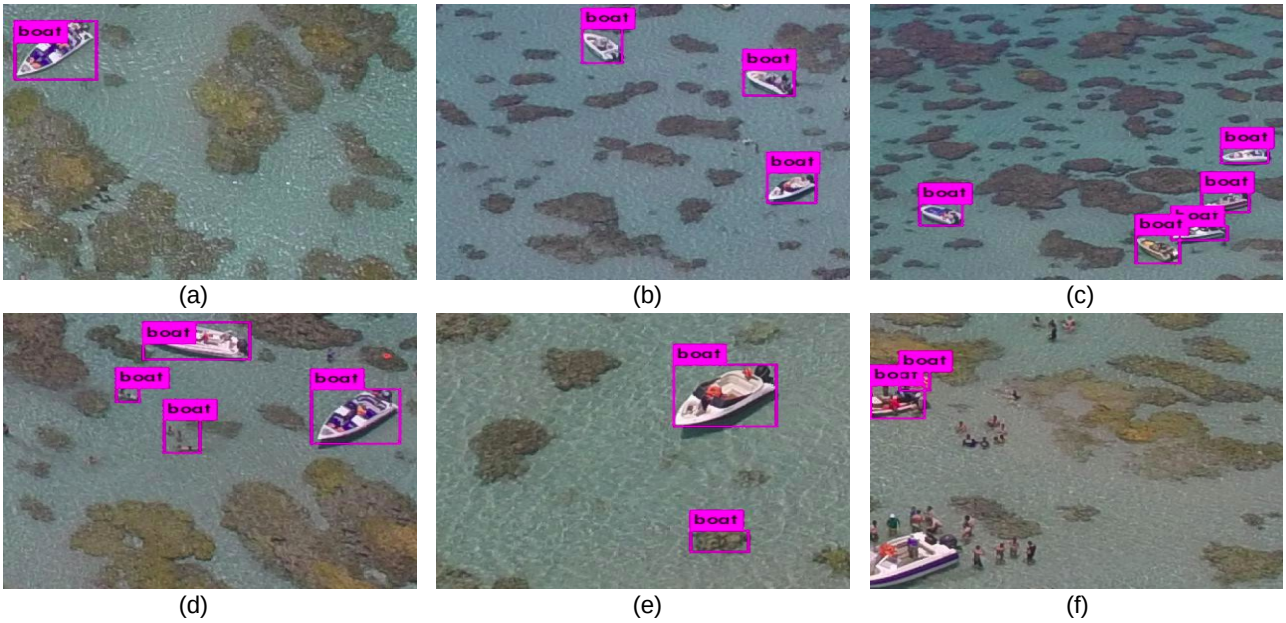


Figure 5. Boat detection examples considering a 0,25 confidence threshold.

To evaluate the developed network, we will analyse the precision, recall and recognition rate ($RR$), shown in Equations 2, 3 and 4, respectively. In short, the precision retrieves the proportion of detected boats that are actually correct while the recall retrieves the proportion of boats that were actually detected. Recognition rate represents the relation between true positives ($TP$) and the total amount of boats ($TB$) in the dataset. Also, we computed the mean average precision ($mAP$), a popular metric for measuring the accuracy of object detectors.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$RR = \frac{TP}{TB} \tag{4}$$

where FN and FP represents false negative and false positive, respectively.

We tested the network performance using different confidence threshold values, as summarized in Table 2. We also considered an Intersection over Union (IoU) threshold of 50%, i.e., if the overlapping area between the predicted and the actual bounding box is more than 50%, the prediction is considered to be correct.

Analyzing Table 2, we can see the best option regarding recall is using a confidence threshold of 0,25 since it provides a recall value of 0,97 with only 6 FN and 190 TP. A threshold value of 0,20 could also be used, but the number of FP would increase by 6 while decreasing only 1 FN. If both precision and recall were considered, a confidence value of 0,45 would be the best case since there are 7 FN with only 15 FP.

Table 2. Network performance for different confidence thresholds. $TB = 219$ boats.

| Thres. | TP | FP | FN | Precision | Recall | Rec. Rate |
|---|---|---|---|---|---|---|
| 0,15 | 191 | 33 | 5 | 0,85 | 0,97 | 0,872 |
| 0,20 | 191 | 31 | 5 | 0,86 | 0,97 | 0,872 |
| 0,25 | 190 | 25 | 6 | 0,88 | 0,97 | 0,867 |
| 0,30 | 189 | 23 | 7 | 0,89 | 0,96 | 0,863 |
| 0,35 | 189 | 19 | 7 | 0,91 | 0,96 | 0,863 |
| 0,40 | 189 | 17 | 7 | 0,92 | 0,96 | 0,863 |
| 0,45 | 189 | 15 | 7 | 0,93 | 0,96 | 0,863 |
| 0,50 | 186 | 12 | 10 | 0,94 | 0,95 | 0,849 |

For the application of this paper, once a boat is detected, the image of this supposed boat will be sent to an operator in the ground station, which will analyze the image and send a team to investigate the occurrence if necessary. Therefore, even if a FP occurs, the operator will be able to analyze the image and choose not to send a team. However, if a FN occurs, there will be human activities in the APARC but the operator would not be notified, which is much more critical. In this sense, we will prioritize a high recall to detect as much human activity as possible and use a confidence threshold of 0,25.

Figure 5 shows some results when considering a 0,25 confidence threshold. In Figures 5a, 5b and 5c, the network correctly detected 1, 3 and 5 boats with accurate bounding boxes, respectively, which illustrates the good performance of the system. An interesting point is that Figure 5d has 2 FP, where the network predicted people as boat. However, since our final goal is to detect human activity, this error is actually helpful. In Figure 5e another FP occurred, where a coral was predicted as boat, and in Figure 5f a FN ocurred, where a boat was not detected.

In our previous work (Santos et al., 2019), we used a SSD network to perform boat detection, where a recognition rate of up to 62,15 % was achieved with 10 FP. In this paper, we significantly improved this metric to 86,70 % while also decreasing the FP to 6. Finally, the network

achieved a mAP of 97,23 % in the test dataset and it took an average time of 15,739 ms to predict each image using the Darknet framework considering a computer with specifications described in the beginning of this section.

## 6. CONCLUSION

In this paper, we presented a novel convolution neural network to detect boats, ships, and other water vehicles located at sea, which will be applied for monitoring human activities in an Environmental Protection Area. Another contribution is the construction of a new testing dataset with fully annotated aerial images of boats.

Experiments proved that the developed neural network achieved outstanding results in the testing dataset, achieving a recognition rate of 86,70%, recall of 97% and precision of 88% considering a confidence threshold of 0,25.

In future works, we will evaluate the performance of the developed network in an embedded system that will be attached to a UAV and test different CNN architectures to improve recognition rate and inference speed. Also, we plan to validate the newly developed network in real-life experiments using a multi-UAV communication system in the APARC.

## ACKNOWLEDGEMENTS

The images used in this study to evaluate the boat detection were provided by Daniel Maciel - IDEMA.

## REFERENCES

Antonio-Javier Gallego, Antonio Pertusa, P.G. (2018). Automatic ship classification from optical aerial images with convolutional neural networks. *Remote Sensing*, 10(4). doi:10.3390/rs10040511.

Bo, L., Xiaoyang, X., Xingxing, W., and Wenting, T. (2021). Ship detection and classification from optical remote sensing images: A survey. *Chinese Journal of Aeronautics*, 34(3), 145–163.

Bochkovskiy, A., Wang, C.Y., and Liao, H.Y.M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

Chen, K., Wu, M., Liu, J., and Zhang, C. (2020a). Fgsd: A dataset for fine-grained ship detection in high resolution satellite images. *arXiv preprint arXiv:2003.06832*.

Chen, Z., Chen, D., Zhang, Y., Cheng, X., Zhang, M., and Wu, C. (2020b). Deep learning for autonomous ship-oriented small ship detection. *Safety Science*, 130, 104812.

Fascista, A. (2022). Toward integrated large-scale environmental monitoring using wsn/uav/crowdsensing: A review of applications, signal processing, and future perspectives. *Sensors*, 22(5), 1824.

Fu, Z., Mao, Y., He, D., Yu, J., and Xie, G. (2019). Secure multi-uav collaborative task allocation. *IEEE Access*, 7, 35579–35587.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580–587.

Gąsienica-Józkowy, J., Knapik, M., and Cyganek, B. (2021). An ensemble deep learning method with optimized weights for drone-based water rescue and surveillance. *Integrated Computer-Aided Engineering*, 1–15. doi:10.3233/ICA-210649.

Hou, X., Ma, J., and Zang, S. (2021). Airborne infrared aircraft target detection algorithm based on yolov4-tiny. In *Journal of Physics: Conference Series*, volume 1865, 042007. IOP Publishing.

Li, H., Deng, L., Yang, C., Liu, J., and Gu, Z. (2021). Enhanced yolo v3 tiny network for real-time ship detection from visual image. *IEEE Access*, 9, 16692–16706.

Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., and Berg, A.C. (2016). Ssd: Single shot multibox detector. In *European conference on computer vision*, 21–37. Springer.

Lodeiro-Santiago, M., Caballero-Gil, P., Aguasca-Colomo, R., and Caballero-Gil, C. (2019). Secure uav-based system to detect small boats using neural networks. *Complexity*, 2019.

Lopes, R.M.R., Soares, I., and De Araújo, J. (2014). Área de proteção ambiental dos recifes de corais–área dos parrachos de maracajaú/rn: desafios para o uso sustentável. *Caminhos Geogr*, 15, 51.

MakeML (2020). Ships dataset. URL `https://makeml.app/datasets/ships`.

Nie, G.H., Zhang, P., Niu, X., Dou, Y., and Xia, F. (2017). Ship detection using transfer learned single shot multi box detector. In *ITM web of conferences*, volume 12, 01006. EDP Sciences.

Redmon, J. (2013–2022). Darknet: Open source neural networks in c. `http://pjreddie.com/darknet/`.

Santos, V.G., Pereira, D.S., Alsina, P., Fernandes, D., Nascimento, L., Leite, D.L., Morais, M.R., Silva, M.R., and Souza, E.S. (2019). Multi-uav system architecture for environmental protection area monitoring. *Proc. Anais do Simpósio Brasileiro de Automação Inteligente*, 1–6.

Silva, L.A., Sanchez San Blas, H., Peral García, D., Sales Mendes, A., and Villarubia González, G. (2020). An architectural multi-agent system for a pavement monitoring system with pothole recognition in uav images. *Sensors*, 20(21), 6205.

Wang, Y., Wang, C., Zhang, H., Dong, Y., and Wei, S. (2019). Automatic ship detection based on retinanet using multi-resolution gaofen-3 imagery. *Remote Sensing*, 11(5), 531.

Yang, G., Li, B., Ji, S., Gao, F., and Xu, Q. (2013). Ship detection from optical satellite images based on sea surface analysis. *IEEE Geoscience and Remote Sensing Letters*, 11(3), 641–645.

Yanmaz, E., Yahyanejad, S., Rinner, B., Hellwagner, H., and Bettstetter, C. (2018). Drone networks: Communications, coordination, and sensing. *Ad Hoc Networks*, 68, 1–15.

Zhang, Y., Guo, L., Wang, Z., Yu, Y., Liu, X., and Xu, F. (2020). Intelligent ship detection in remote sensing images based on multi-layer convolutional

feature fusion. *Remote Sensing*, 12(20), 3316.

Zhu, D., Xu, G., Zhou, J., Di, E., and Li, M. (2021). Object detection in complex road scenarios: Improved yolov4-tiny algorithm. In *2021 2nd Information Communication Technologies Conference (ICTC)*, 75–80. IEEE.

Zou, Z. and Shi, Z. (2016). Ship detection in spaceborne optical image with svd networks. *IEEE Transactions on Geoscience and Remote Sensing*, 54(10), 5832–5845.