UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

Shark Tracking Using Deep Learning

Alvaro Francisco Peña Solis

Ingeniería en Sistemas

Trabajo de fin de carrera presentado como requisito
para la obtención del título de
Ingeniero de Sistemas

Quito, 07 de Mayo de 2020

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

HOJA DE CALIFICACIÓN
DE TRABAJO DE FIN DE CARRERA

Shark tracking using deep learning

Alvaro Francisco Peña Solis

Nombre del profesor, Título académico **Noel Pérez, Ph.D.**

Quito, 7 de mayo de 2020

DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído todas las Políticas y Manuales de la Universidad San Francisco de Quito USFQ, incluyendo la Política de Propiedad Intelectual USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Firma del estudiante: _____

Nombres y apellidos: Alvaro Francisco Peña Solis

Código: 00133680

Cédula de identidad: 1805429139

Lugar y fecha: Quito, Mayo de 2020

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en http://bit.ly/COPETheses.

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on http://bit.ly/COPETheses.

RESUMEN

El tiburón martillo festoneado (Sphyrna lewini) fue clasificado recientemente como una especie en peligro crítico en la lista roja de la UICN. A pesar de las disminuciones mundiales, hay una falta de información sobre la situación de esta especie en el Pacífico oriental tropical, en parte debido a la falta de una vigilancia independiente de las pesquerías. El uso de material de vídeo puede ser una herramienta valiosa para desarrollar indicadores estandarizados, pero el análisis de las imágenes puede ser muy laborioso. En este estudio, proponemos un nuevo método automatizado basado en redes neuronales convolucionales profundas para detectar y rastrear en secuencias de vídeo a los amenazados tiburones martillo. El método propuesto mejoró la arquitectura estándar de YOLOv3 deep, añadiendo 18 capas más (16 capas convolucionales y 2 capas de Yolo), lo que aumentó el rendimiento del modelo en la detección de las especies bajo análisis a diferentes escalas. Según la validación basada en el análisis de los fotogramas, el método propuesto superó la arquitectura estándar de YOLOv3 en cuanto a las puntuaciones de precisión para la mayoría de los fotogramas inspeccionados. Además, la media de precisión y recordamiento en un conjunto de fotogramas experimentales formado mediante el método de validación cruzada de 10 veces manifestó que el método propuesto era mejor que la arquitectura estándar de YOLOv3, alcanzando puntuaciones de 0.99 y 0.93 frente a 0.95 y 0.89 para la media de precisión y recordamiento, respectivamente. Además, ambos métodos pudieron evitar la introducción de detecciones positivas falsas, pero no pudieron resolver el problema de la oclusión de especies. Nuestros resultados indican que el método propuesto es una herramienta alternativa viable que podría ayudar a vigilar la abundancia relativa de los tiburones martillo en la naturaleza.

Palabras clave: Seguimiento y detección de tiburón martillo, detector en tiempo real, red neuronal profunda, arquitectura YOLOv3

ABSTRACT

Scalloped hammerhead sharks (Sphyrna lewini) were recently classed as Critically Endangered on the IUCN Red List. Despite global declines, there is a lack of information on the status of this species in the Eastern Tropical Pacific, partly due to inconsistent fisheries-independent monitoring. The use of video footage can be a valuable tool to develop standardized indicators, yet analysis of footage can be highly laborious. In this study, we propose a new automated method based on deep convolutional neural networks to detect and track endangered hammerhead sharks in video sequences. The proposed method improved the standard YOLOv3deep architecture by adding 18 more layers (16 convolutional and 2 Yolo layers), which increased the model performance in detecting the species under analysis at different scales. According to the frame analysis based validation, the proposed method outperformed the standard YOLOv3 architecture in terms of accuracy scores for the majority of inspected frames. Also, the mean of precision and recall on an experimental frames dataset formed using the 10-fold cross-validation method highlighted that the proposed method was better than the standard YOLOv3 architecture, reaching scores of 0.99 and 0.93 versus 0.95 and 0.89 for the mean of precision and recall, respectively. Furthermore, both methods were able to avoid introducing false positive detections.However, they were unable to handle the problem of species occlusion. Our results indicate that the proposed method is a feasible alternative tool that could help to monitor relative abundance of hammerhead sharks in the wild.

Keywords: Hammerhead shark detection and tracking, real-time detector, deep convolutional neural network, YOLOv3 architecture.

AGRADECIMIENTOS

Agradezco a mi padre y madre por su apoyo incondicional durante toda la carrera.

Le doy gracias a Noel Perez por guiarme durante este proceso.

DEDICATORIA

A mi padre y madre por animarme a estudiar esta valiosa carrera.

Alvaro.

TABLA DE CONTENIDO

INDICE DE TABLAS

INDICE DE FIGURAS

INTRODUCTION

Object detection and tracking play an important role in real world applications such as: surveillance (Raghunandan et al., 2018), aiding people with physical disabilities (Dionisi, Sardini, & Serpelloni, 2012), microscopic examination (Wang et al., 2019) and marine species analysis (Xu, Bennamoun, An, Sohel & Boussaid, 2019). Monitoring of marine species has been carried out widely during the past decade, but the associated analytical tasks rely heavily on the biologists, which could introduce errors by the manual process. Implementing automated detection and tracking systems can mitigate these errors by reducing human interaction with the environment and providing a second opinion tool for biologists on a range of applications.

Advances in machine learning topics and especially deep learning using convolutional neural networks (CNN) are significant in object detection (Liu et al., 2016), (Redmon, Divvala, Girshick & Farhadi, 2016), (Redmon & Farhadi, 2017), (Redmon & Farhadi, 2018), (Voulodimos, Doulamis, Doulamis & Protopapadakis, 2018) and (O'Mahony et al., 2019), where they have proven to outperform traditional machine learning methods in accuracy and speed metrics. These improvements make such algorithms favorable for using them in real-world applications.

Automated marine species detection and tracking constitute a vital area of application due to the need to track the population status of threatened and endangered species in the aquatic ecosystem. In (Maire, Alvarez & Hodgson, 2015), a method based on region segmentation was proposed, which included deep convolutional neural networks (CNN) to improve the recall and precision metrics in detecting marine mammals. The method was tested using a dataset of aerial images retrieved from wildlife surveys. In (Xu et al.,

2019), a study using more complex computer vision techniques in conjunction with deep CNN models was proposed to detect and classify different species of fish. In (Uemura, Lu & Kim, 2020), the YOLO method was implemented to detect and track marine organisms, including sharks.The method obtained satisfactory results when it was tested in deep-sea videos. Furthermore, in (Raza, & Hong, 2020), an improved version of the YOLOv3 method was proposed for detecting fish and sharks, which overcame the standard method in the mean of precision score performance.

One shark in particular, lends itself to the development of species recognition techniques due to its unique body shape. The scalloped hammerhead shark (Sphyrna lewini) is a medium sized coastal-pelagic shark that can attain a size of over 4 m (but usually not more than 2-3 m) (Rigby et al., 2019). It has a circumglobal distribution and is thought to be divided into several genetically discrete populations, among which the Eastern Pacific population (from Baja California (USA) to northern Peru) is under considerable threat from fishing activity, and is the main source of hammerhead shark fins in Hong Kong markets (Fields, Fischer, Shea, Zhang, Feldheim, & Chapman, 2020). Hammerhead sharks, along with all other shark species, are not officially targeted in countries such as Costa Rica and Ecuador, yet a legal loophole allowing for the sale of sharks caught as ”by-catch” has resulted in at least 200,000 sharks landed each year in Ecuador alone (Hearn & Bucaram, 2017), (Martinez-Ortiz, Aires-da Silva, Lennert-Cody & Maunder, 2015). Both Ecuador and Costa Rica have made efforts to protect their marine biodiversity, notably the creation of marine protected areas (MPAs) around their oceanic islands of Galapagos and Cocos respectively. However, scientists have found that hammerheads migrate between the reserves,

becoming vulnerable to fishing pressure once they leave protected waters (Hearn et al., 2014).

In late 2019, the red listing status for the species as a whole was amended from ”Endangered” to ”Critically Endangered” (Rigby et al., 2019). As yet, neither reserve has a formal process for evaluating the population trends for sharks, but diver observations over a > 20 year period at Cocos Island suggested severe declines in numbers (Peñaherrera-Palma et al., 2018), while a study of dive guide perceptions in the Galapagos Islands obtained similar results (Peñaherrera-Palma et al., 2018). There is a need to develop low cost, standardized tools to evaluate their trends in reserves where fishing is not permitted, and thus landings data not an option. In recent years, several tools have been developed which involve the use of video footage, either operated by SCUBA divers or remotely (White, Myers, Flemming & Baum, 2015), (Acuña-Marrero, Smith, Salinas-de-León, Harvey, Pawley & Anderson, 2018), (Bouchet & Meeuwig, 2015). However, the analysis of the resulting footage can be labor-intensive and would benefit greatly from automation.

The study of marine species has many problems such as object occlusion, blurring, poor lighting conditions, focus variations to the object, and projection against the sunlight. Despite the recent advances in machine learning applied to the marine environment, detecting, classifying and tracking marine species remain challenging to tackle because of the uncontrolled environment associated with these tasks.

In this study, we propose a new automated method based on a deep CNN architecture to detect and track hammerhead sharks in video sequences recorded at the Galapagos and Coco Islands. The proposed method improved the standard YOLOv3 deep

architecture (Redmon & Farhadi, 2018) by including 18 more layers, which increased the model performance in detection and tracking of the species under analysis. With this approach, the biology research community could have a viable tool to help them analyze this shark species.

The remainder of this paper is organized as follows: the Materials and Methods section, presents the hammerhead shark video database used for our experimentation, a brief description of the standard YOLOv3 deep architecture, the proposed method and the experimental setup designed to evaluate it. The Results and Discussion section presents a head-to-head comparison based on the accuracy (ACC), precision, recall, and mean of loss function scores, between the standard YOLOv3 deep architecture and our proposed method. Finally, conclusions and future work are discussed in the last section.

MATERIALS AND METHODS

Yolov3 framework

This method is a recent deep neural network used for object detection and real-time tracking (Redmon & Farhadi, 2018). Its core consists of a backbone network named Darknet-53 for feature extraction, and YOLO layers for predicting the bounding box of desired objects at three different scales. That means, it is possible to detect little and large objects at the same time, becoming a powerful architecture in the context of object detection.

The Darknet-53 network is composed of residual blocks, containing convolutional layers inside. These blocks serve mainly as feature extractors and since this network needs to explore the whole feature space from block to block, it does not involve any max-pooling layer in its configuration. On the other hand, the YOLO layers are composed of 7 convolutional layers, and 3 upsampling layers between the convolutional ones, to scale up the input RGB (red, green, blue) images with dimensions of (416 x 416 x 3) at each time. A brief description of the YOLOv3 architecture is shown in Fig. 1.

This architecture has demonstrated to be competitive in object recognition against other developed methods. Even though it is considered a heavy architecture that consumes significant resources, it is more efficient than ResNet-101 or ResNet-152; it is three times faster than the SSD (Single Shot Detector neural network) and its variants. Additionally, it is similar in performance to the RetinaNet on the COCO dataset.

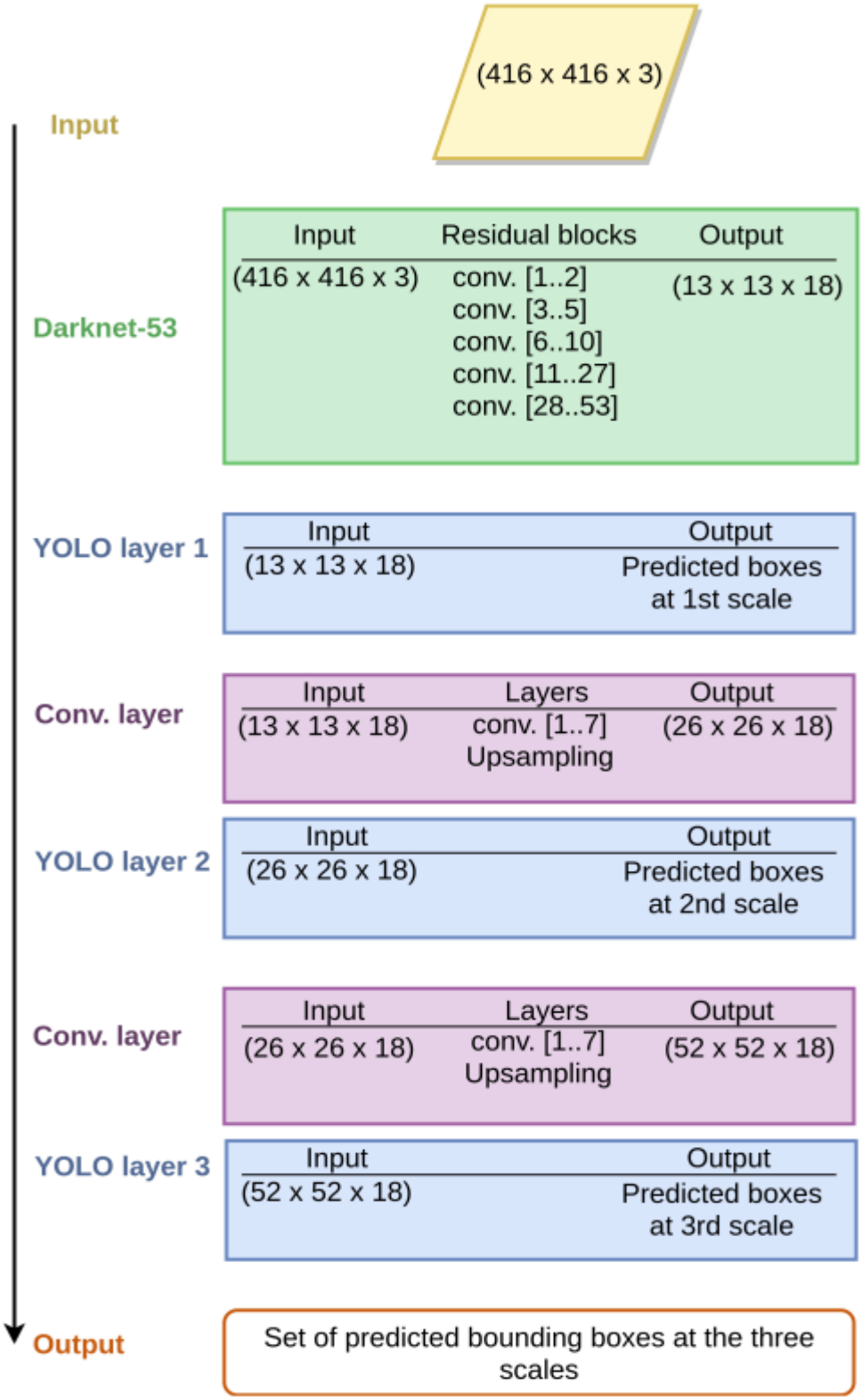

Figure 1 A brief description of Yolov3 architecture

Proposed method

Detecting and tracking marine species, such as hammerhead sharks, is considered to be a challenge. Although the shark silhouette is easy to recognize, there are uncontrolled environmental conditions such as poor lighting, occlusions by non-desired fish species, projection against the sunlight, among others, which make the task difficult. To overcome this, we proposed a new method, which improved the YOLOv3 standard architecture by including 18 more layers. This improvement aims to detect and track the hammerhead sharks species accurately.

An overall perspective of the developed method is shown in Fig. 2. From this, it is possible to notice that our method combined the standard YOLOv3 architecture (see Fig. 1) plus some specific layers designed to tackle the problem under analysis. The major improvement over the standard YOLOv3 architecture was resizing the input images, which passed from (800 x 422 x 3) to (608 x 608 x 3) dimensions. The remainder of the method consists of attaching some layers at the end of the standard YOLOv3 architecture distributed in the following order: 7 convolutional layers as feature extractors, and 1 upsampling layer to scale up the input image size, both inclusions with similar configurations to the standard YOLOv3 architecture, 1 YOLO layer for predicting a set of bounding boxes at the new scale. This structure was repeated one more time to complete the designed architecture, which accomplished a total of 18 added layers.

It should be noted that the YOLO layers in the proposed method were set to perform at the 4th and 5th scales, respectively. Also, the anchor box size on both layers was tuned to be smaller than the one employed by the standard YOLOv3 architecture (see Fig. 2). This property represents the ideal size and location of predicted objects in the image, in this case,

hammerhead sharks. Thus, the better the property adjustment, the better bounding box prediction, independently of the object size. In contrast, this property in the standard YOLOv3 architecture is pre-determined for the COCO dataset (Lin et al., 2014).

Since the standard YOLOv3 architecture is configured to detect medium-large objects, adding these improvements enabled the proposed method to detect smaller objects as well, which increases the model's learning rate and thus improves the real-time detection.

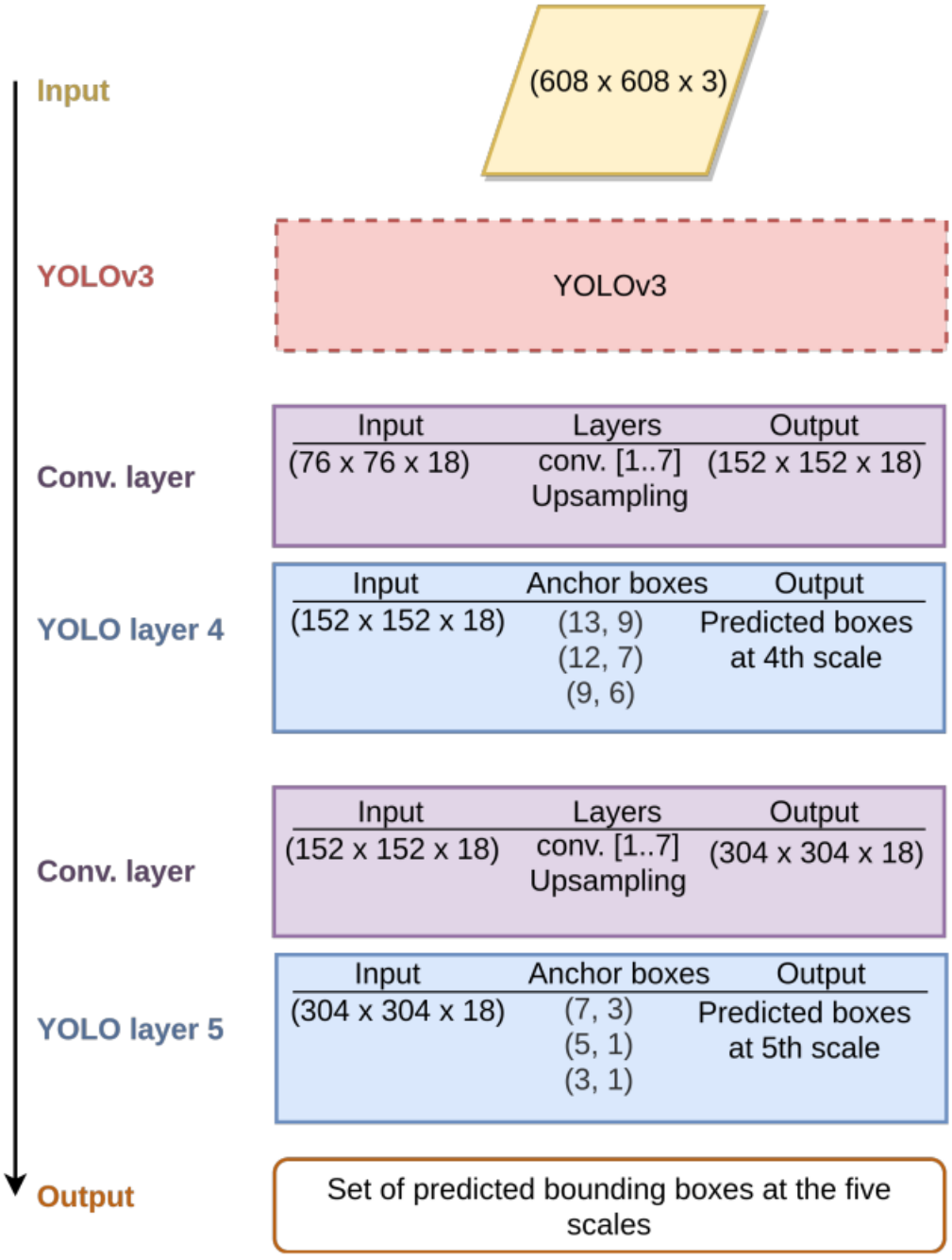

Figure 2 A brief description of the proposed method

Shark database

We used three footage sources as our main shark database. These sources were filmed at the Galapagos [0˝ 39’ 59.99” N´ 90˝ 32’ 59.99” W] and Coco [5˝ 31’ 4.79” N ´87˝ 04’ 10.80” W] Islands, both located in the Eastern Tropical Pacific Ocean and administered by Ecuador and Costa Rica, respectively. Both sets of islands are surrounded by marine reserves and are UNESCO World Heritage Sites due to their outstanding biodiversity, including large aggregations of several shark species.

The video footage used in this study is mostly of scalloped hammerhead sharks, but other marine species, including other sharks, also feature in the same video samples. The duration of each sample varied between 30 to 50 seconds, the recording format was file.mp4 at 24 fps (frames per second) and they were taken by biologists in uncontrolled environments. That means, the sharks are far from the camera lens, the illumination is poor and the projection view is against the sunlight most of the time, thus we used footage that might be considered typical quality from non-professional film crew with underwater cameras.

Experimental setup

This section describes the experimental setup created to validate the proposed deep learning architecture. The video preprocessing and dataset creation, training and test partitions, anchor box values and validation metrics are aspects to be presented next.

Video pre-processing and dataset creation

This step aims to provide the needed samples of hammerhead shark species to form an experimental dataset that will serve to train and test the proposed method. Thus, for all videos

in the database, we applied a decoding operation to extract all the frames contained by the video source by using the ffmpeg framework (FFmpeg developers, 2016). Eachvideo sequence of 50s of duration at 24fps provided 1200 frames. However, we discarded around 50% of frames by removing those who are too blurry or contain species occlusions. After processing the videos, we gathered a total number of 1012 valid frames.

Since the number of collected frames does not fulfill the need to have enough samples for training deep learning models without incurring on overfitting, a data augmentation technique (Curilem, Canário, Franco & Rios, 2018) was applied to increase the number of frames containing hammerhead sharks. Thus, each frame was rotated by 30, 45 and 210 degrees to form an experimental dataset containing a total number of 2000 frames with dimensions of [800 x 422]. Besides, a labeling operation was carried out on the frames tomark the regions that belong or not to the hammerhead shark class. This operation provided an annotation file, in which each row contains information about the bounding box and output class label of each marked region within the frame. Both the experimental dataset of frames and its corresponding annotation file are mandatory to train the standard YOLOv3 model and thus, the proposed method.

Training and test partitions

We applied the stratified 10-fold cross-validation method (Purushotham & Tripathy, 2011) to build disjoint training and test partitions. In this way, the proposed method is trained using different training sets, which enable it to learn from different input space representations. Testing on these different sets encourages the resulting variability in the classification of individual samples. The use of this method helps to avoid overfitting.

Anchor box values

These values were determined experimentally by observing the smallest hammerhead sharks of interest across all images (frames) of the experimental dataset. This process allowed us to estimate the dimensions (in pixels) of observed samples. With this information, the objectness score parameter was computed, which was set to the YOLO layers in the proposed architecture. The objectness score manages whether or not found hammerhead shark objects are presented in the frame under analysis (Christiansen, 2018).

Validation metrics

A video source that was not considered during the model's training step was used to test the proposed method in real-time. The performance of the method was based on the accuracy (ACC) of hammerhead sharks detection and tracking across a set of retrieved frames of the test video. A variation of this validation protocol was previously used in (Sung, Yu & Girdhar, 2017) to assess fish detection in real-time. Thus, we established a three-step procedure for conducting the evaluation:

Selecting nine frames (empirical selection) in the test video starting at time 0 to the video duration (vd) with an increment factor determined by the following splitting time (sp) formula: $sp = truncate(\frac{vd}{9})$. Counting the number of correct hammerhead shark detections out of the total presented in the current frame under analysis. Tracking the hammerhead sharks by counting how many of them were correctly detected across all the inspected frames.

Additionally, for the head-to-head comparison between the proposed method and the standard YOLOv3 architecture, we computed the mean value of the precision, recall, and loss function, using the 10-fold cross-validation method in the training-test steps.

All implementations were done in Python language version 3.5 (Van Rossum & Drake, 2009) with the scikit-learn (SKlearn) (Pedregosa et al., 2011), Pytorch version 0.4 (Paszke, 2019), OpenCVversion 4.0.2.32 (Bradski, 2000), Numpy (Oliphant, 2006) and scikit-image (SKimage) (Walt et al., 2014) libraries, and using Darknet (Redmond, 2013) as backend.

RESULTS AND DISCUSSIONS

According to the experimental setup section, we validated the detection performance of the proposed method in a real-time scenario by analyzing nine frames recovered from the employed test video. Also, the head to head comparison against the standard YOLOv3 architecture was made using the 10-fold cross-validation method applied to the experimental dataset of frames. The obtained ACC, mean of precision, recall, and loss function scores revealed interesting results in detecting hammerhead sharks.

Performance of the proposed method

Regarding the detection performance of hammerhead sharks using the frames analysis, the proposed method was able to detect the target species with ACC scores above 50% for the majority of inspected frames as it is shown in Table I. Onlythe frames with ID 4, 5, and 7 provided a lower ACC score of detection. These results could be explained by the filming conditions associated with the marine environment, where camera movements and projections against the sunlight are common issues. In all videos of the database, the hammerhead sharks performed random trajectories by approaching and moving away from the camera lens. This behavior provoked either the distortion or blurring of the targets and, consequently, the failure of detection.

Two additional factors contributed to non-detection of sharks: the partial shape of the shark in the frame, and the target occlusion by other marine species (see Fig. 4). For example, at the top of the frames with ID 1 and 3 (Fig. 4 top row), there was one hammerhead shark showing half of its silhouette. Although it was close to the camera lens

like other sharks, which were captured in the frame, this one was not considered by the proposed method. In terms of occlusions, the proposed method failed to detect several hammerhead sharks in the range of frames with ID from 4 to 8 because fishes occluded them. However, in the frame with ID 3, one hammerhead shark was identified without taking into consideration the other closest fish (see Fig.4), (frame ID 3, middle-right target). This situation occurs when sharks look bigger than fishes. In contrast, when fishes look similar in size than the sharks, the detector was not activated like in the frame with ID 6 (see Fig. 4, at the center), which is a good sign of performance. Similarly, the fishes in the frames with ID 7 and 8 are in between the camera lens and the hammerhead sharks, but the detector focused only on the sharks while ignoring the fishes (see Fig. 4). Finally, in the last frame with ID 9, there was only the presence of a fish. As it was expected, the proposed method did not record any detections. Thus, it did not introduce false-positive detection on any of the inspected frames, which is an excellent detection performance.

Frame (ID)	Time (s)	Number of sharks per frame (u)	Correct detection		ACC based detection (%)	
			YOLOv3	Proposed method	YOLOv3	Proposed method
1	2	11	0	7	0	64
2	4	9	0	6	0	67
3	6	10	0	6	0	60
4	8	10	2	2	20	20
5	10	7	2	3	29	43
6	12	6	3	4	50	66
7	14	4	1	1	25	25
8	16	4	2	2	50	50
9	18	0	0	0	100	100

ACC - accuracy; *values rounded to the closest integer

Table I Comparison based on the acc per frame between the proposed method and implemented Yolov3 architecture

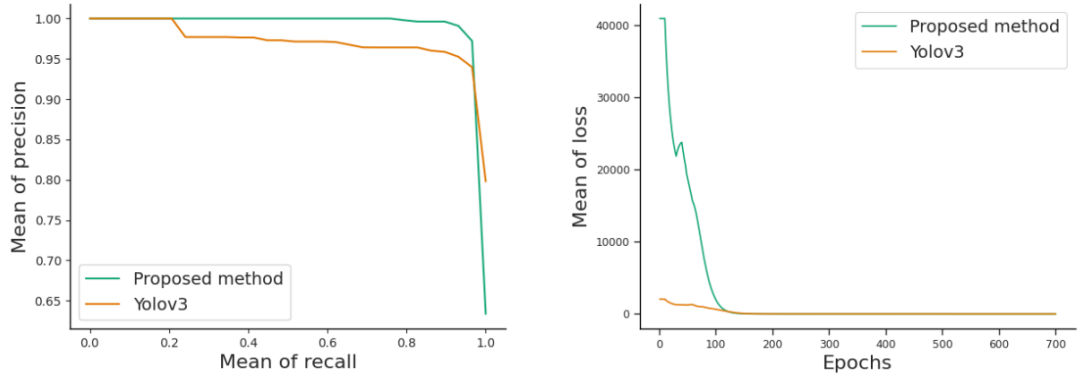

Figure 3 Performance of the proposed method and the standard YOLOv3 architecture in terms of the mean of precision and recall (left) and mean of the lossfunction (right) over ten folds

Head-to-head comparison against the YOLOv3 framework

Concerning the detection of hammerhead sharks using the frames validation, the performance results on both implemented methods can be seen in Table I. From this table, it is possible to see that the proposed method outperformed the standard YOLOv3 architecture. These results could be related to the internal configuration of each method. The proposed method added 18 more layers, including convolutional units (convolutional and upsampling layers), and two Yolo layers for predicting bounding boxes at scales fourth and fifth, which are missing in the standard YOLOv3 architecture. The inclusion of these layers enabled the proposed method to detect hammerhead sharks of different sizes. For example, by analyzing the first three frames, the proposed method was able to detect 19 versus 0 (by the YOLOv3 architecture) out of 30 hammerhead sharks presented on those frames. However, both methods were unable to overcome the problem of occlusions by other sub aquatic species. A visual comparison between both methods on the nine recovered frames of the test video is shown in Fig. 4 and 5.

We also compared both methods by analyzing the mean of precision and recall metrics using the 10-fold cross-validation method on the experimental frames dataset. The precision measured the model's ability to predict the shark bounding boxes correctly. Meanwhile, the recall provided the model's importance to detect the sharks in the frames appropriately. Thus, the higher the precision and recall scores, the better performance of the model. The obtained results, according to both metrics, are shown in Fig. 3, left plot. From this figure, we can state that the precision and recall values of 0.99 and 0.93 reached by the proposed method were superior to the precision and recall scores of 0.95 and 0.89 attained by standard YOLOv3 architecture.Further, neither model incurred in

overfitting during the training processes. The mean of the loss function of both methods decreased across the epochs to meet the learning rate value, as can be seen in Fig. 3, right plot.

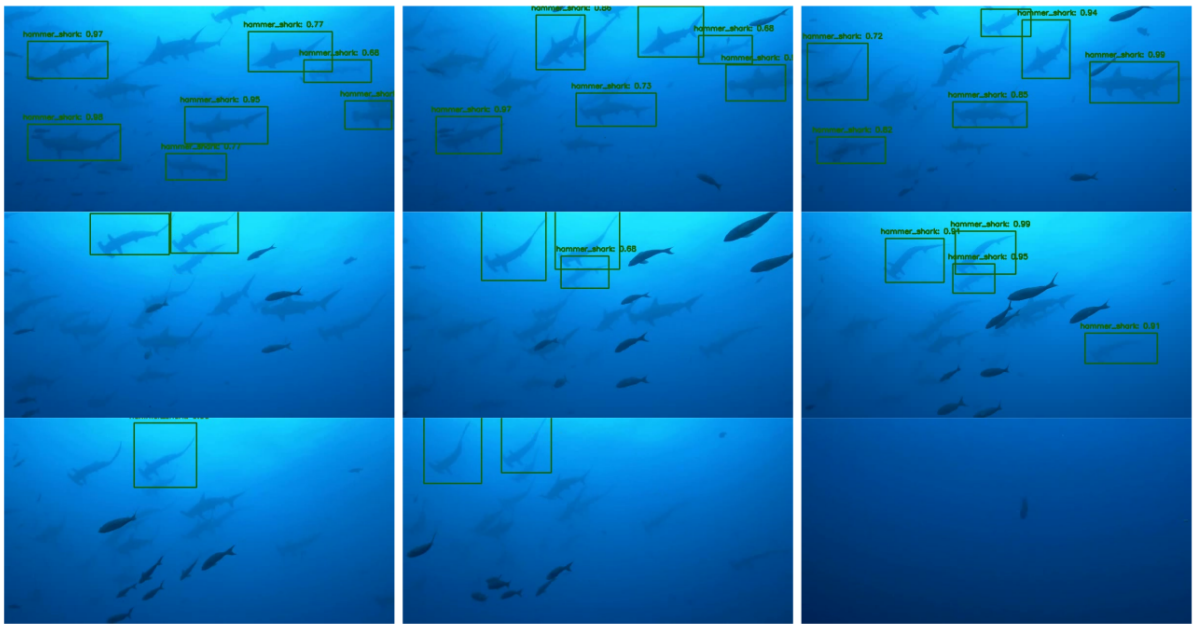

Figure 4 Performance of the proposed method across the frames under analysis: successfully (green box) hammer shark detection in a test video

Figure 5 Performance of the YOLOv3 method [8] across the frames under analysis: successfully (green box) hammer shark detection in a test video.

CONCLUSIONS AND FUTURE WORK

In this study, we developed a new automated method based on deep CNN architecture to detect and track hammerhead sharks in video sequences recorded at the Galapagos and Cocos Islands marine protected areas. The proposed method improved the standard YOLOv3 deep architecture [8] by including 18 more layers (convolutional and Yolo layers), which increased the model performance in detecting the species under analysis at different scales. According to the frame based validation analysis,the proposed method outperformed the standard YOLOv3 architecture in terms of ACC scores for the majority of inspected frames. Concerning the mean of precision and recall on an experimental dataset of frames constructed using the 10-foldcross-validation method, the proposed method was better than the standard YOLOv3 architecture, reaching scores of 0.99 and0.93 versus 0.95 and 0.89 for the mean of precision and recall, respectively. It should be stated that both methods were able to avoid introducing false positive detections. However, they were unable to handle the problem of species occlusion. These results provided clear evidence that the proposed method improved the hammerhead sharks detection while outperforming the standard YOLOv3 architecture, enabling it as a feasible alternative tool to help the analysis of this shark species in the wild

REFERENCES

Acuña-Marrero, D., Smith, A., Salinas-de-León, P., Harvey, E., Pawley, M., & Anderson, M. (2018). Spatial patterns of distribution and relative abundance of coastal shark species in the Galapagos Marine Reserve. Marine Ecology Progress Series. (pp. 73–95).

Bouchet, P., & Meeuwig, J. (2015). Drifting baited stereo-videography: a novel sampling tool for surveying pelagic wildlife in offshore marine reserves. Ecosphere, 6(8), art137.

Bradski, G. (2000). The OpenCV LibraryDr. Dobb's Journal of Software Tools.

Christiansen, A. (2018, October 15). Medium: Anchor Boxes — The key to quality object detection. x https://bit.ly/3cj36ZV.

Dionisi, A., Sardini, E., & Serpelloni, M. (2012). Wearable object detection system for the blind. In 2012 IEEE International Instrumentation and Measurement Technology Conference Proceedings (pp. 1255–1258).

FFmpeg Developers. (2016). ffmpeg tool (Version be1d324) [Computer Software]. Available:http://ffmpeg.org/

Fields, A., Fischer, G., Shea, S., Zhang, H., Feldheim, K., & Chapman, D. (2020). DNA Zip-coding: identifying the source populations supplying the international trade of a critically endangered coastal shark. Animal Conservation. Available:https://zslpublications.onlinelibrary.wiley.com/doi/abs/10.1111/acv.12585.

Hearn, A., & Bucaram, S. (2017). Ecuador's sharks face threats from withinScience, 358(6366), 1009.

Hearn, A., Acuña, D., Ketchum, J., Peñaherrera, C., Green, J., Marshall, A., Guerrero, M. & Shillinger, G. (2014). Elasmobranchs of the Galapagos MarineReserve. Cham: Springer International Publishing. (pp. 23–59).

Joseph Redmon. (2013). Darknet: Open Source Neural Networks in C. Available:http://pjreddie.com/darknet/.

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. (2014). Microsoft coco: Common objects in context. In European conference on computer vision (pp. 740–755).

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

M. Curilem, J. P. Canário, L. Franco, & R. A. Rios (2018). Using CNN To Classify Spectrograms of Seismic Events From Llaima Volcano (Chile). In 2018 International Joint Conference on Neural Networks (IJCNN) (pp. 1-8).

Maire, F., Alvarez, L., & Hodgson, A. (2015). Automating marine mammal detection in aerial images captured during wildlife surveys: a deep learning approach. In Australasian Joint Conference on Artificial Intelligence (pp. 379–385).

Martinez-Ortiz, M., Aires-da Silva, M., Lennert-Cody, C. & Maunder, M. (2015). The Ecuadorian Artisanal Fishery for Large Pelagics: Species Composition and Spatio-Temporal Dynamics. PLOS ONE. (pp. 1-29).

O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G., Krpalkova, L., Riordan, D., & Walsh, J. (2019). Deep learning vs. traditional computer vision. In Science and Information Conference (pp. 128–144).

Oliphant, T. (2006). NumPy: A guide to NumPy. Available:https://numpy.org/

Paszke, A., Gross, A., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., & Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. in Advances in Neural Information Processing Systems. (pp. 8024–8035). Available:http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python Journal of Machine Learning Research, 12, 2825–2830.

Peñaherrera-Palma, C., van Putten, I., Karpievitch, Y., Frusher, S., Llerena-Martillo, Y., Hearn, A. & Semmens, J. (2018). Evaluating abundance trends of iconic species using local ecological knowledge. Biological Conservation. (pp. 197 - 207).

Purushotham, S., & Tripathy, B. (2011). Evaluation of classifier models using stratified tenfold cross validation techniques. In International Conference on Computing and Communication Systems (pp. 680–690).

Raghunandan, A., Raghav, P., Aradhya, H., et al. (2018). Object detection algorithms for video surveillance applications. In 2018 International Conference on Communication and Signal Processing (ICCSP) (pp. 0563–0568).

Raza, K., & Hong, S. (2020). Fast and Accurate Fish Detection Design with Improved YOLO-v3 Model and Transfer Learning. In International Journal of Advanced Computer Science and Applications.

Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7263–7271).

Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvementarXiv preprint arXiv:1804.02767.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779–788).

Rigby, C., Dulvy, N., Barreto, R., Carlson, J., Fernando, D., Fordham, S., Francis, M., Herman, K., Jabado, R., Liu, K., & others. (2019). Sphyrna lewini. The IUCN Red List of Threatened Species 2019: e. T39385A2918526.

Sung, M., Yu, S. & Girdhar. Y. (2017). Vision based real-time fish detection using convolutional neural network. In OCEANS 2017 - Aberdeen (pp. 1-6).

Uemura, T., Lu, H., & Kim, H. (2020). Marine organisms tracking and recognizing using yolo. In 2nd EAI International Conference on Robotic Sensor Networks (pp. 53–58).

Van Rossum, G., & Drake, F. L. (2009). Python 3 Reference Manual. Scotts Valley, CA: CreateSpace.

Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep learning for computer vision: A brief review Computational intelligence and neuroscience.

Walt, ., Schönberger, ., Boulogne, ., Warner, ., Yager, ., Gouillart, ., & the scikit-image contributors (2014). scikit-image: image processing in PythonPeerJ, 2, e453.

Wang, Q., Bi, S., Sun, M., Wang, Y., Wang, D., & Yang, S. (2019). Deep learning approach to peripheral leukocyte recognitionPloS one, 14(6).

White, E., Myers, M., Flemming, J., & Baum, J. (2015). Shifting elasmobranch community assemblage at Cocos Island--an isolated marine protected area. Conservation Biology, 29(4), 1186-1197.

Xu, L., Bennamoun, M., An, S., Sohel, F., & Boussaid, F. (2019). Deep learning for marine species recognition. In 2019 Springer Handbook of Deep Learning Applications. (pp. 129–145).