

Desarrollo de un sistema basado en software para la identificación automática de especies marinas



Colaboración

Marco Antonio Vivar Olvera, Benemérita Universidad Autónoma de Puebla; Eduardo Vázquez Santacruz, Universidad Autónoma Metropolitana

Fecha de recepción: 16 de noviembre de 2022

Fecha de aceptación: 21 de diciembre de 2022

RESUMEN: Hasta el momento, se han utilizado diferentes métodos de identificación y reconocimiento biológico, que en su gran mayoría refiere al uso de técnicas invasivas. En este proyecto nos centramos en trabajar con especies marinas, de tal forma que se busca implementar un sistema de identificación automático de tortugas y peces basado en algoritmos de Inteligencia Artificial con el objetivo de coadyuvar en la conservación y uso de forma sostenible de los recursos marinos.

Para la construcción de este primer prototipo del sistema referido se usaron clasificadores Haar Cascades, método definido por Viola y Jones, el cual es un enfoque clásico basado en aprendizaje de máquinas utilizado para detectar objetos, en donde el entrenamiento se realiza con imágenes positivas y negativas. A grandes rasgos el proceso que se siguió para el desarrollo del sistema fue preparar el conjunto de datos de entrenamiento, posteriormente con base en las muestras de este conjunto referido se entrenó el modelo del clasificador en cascada, finalmente se utilizó el clasificador para ejecutar nuestro sistema basado en software de inteligencia artificial para la detección automática de especies marinas, específicamente tortugas y peces.

PALABRAS CLAVE: aprendizaje automático, clasificadores haar cascade, detección de objetos, inteligencia artificial, visión artificial.

ABSTRACT: So far, different methods of identification and biological recognition have been used, where the vast majority refer to the use of invasive techniques. In this project we focus on working with marine species, in such a way that it seeks to implement a system of automatic identification of turtles and fish based on Artificial Intelligence algorithms to support the cause of conserving and sustainably using marine resources.

Haar Cascades classifiers, a method defined by Viola and Jones, which is a classic approach based on machine learning, where training is done with positive and negative images. In general the process that was followed by us for the development of the system was to prepare the training data set, then based on the samples of this referred set the model of the cascading classifier was trained, finally the classifier was used to run our system based on artificial intelligence software for automatic detection of marine species, specifically turtles and fish.

KEYWORDS: artificial intelligence, haar cascade classifier, object detection, machine learning, machine vision.

INTRODUCCIÓN

La presente investigación brinda una solución tecnológica para contribuir con el propósito catorce de los objetivos de desarrollo sostenible (ODS), los cuales fueron adoptados por las Naciones Unidas en 2015 como un llamamiento universal para poner fin a la pobreza, proteger el planeta y garantizar que para el año 2030 todas las personas disfruten de paz y prosperidad [1]. Esta solución en prototipo que sugerimos consiste en un sistema de reconocimiento automático de especies marinas, el cual también podrá ayudar en el monitoreo no invasivo de la biodiversidad a través de algoritmos de aprendizaje de máquinas y otros de inteligencia artificial.

Machine Learning o aprendizaje de máquinas es la disciplina dentro del campo de la inteligencia artificial, se relaciona con las subcategorías de la ya mencionada IA; la subcategoría en la que se basa este artículo es la visión y el método utilizado nos ayuda en la detección de objetos, nuestros objetos son las especies marinas.

MATERIAL Y MÉTODOS

Durante el desarrollo de esta investigación se ha requerido el uso de una serie de herramientas que fueron de gran ayuda para la realización del proyecto; las cuales son: Visual Studio Code, Cascade Trainer GUI, Python 3.10.5 y OpenCV.

Solución Propuesta

Para la elaboración del sistema de identificación automática de especies marinas para el caso de tortugas y peces, tenemos dos opciones para entrenar nuestro modelo inherente, el uso de Haar Cascade o Redes Neuronales Convolucionales (CNN). Las detecciones de Haar Cascades son más rápidas, pero menos precisas, mientras que las de CNN son más lentas, pero más precisas.

En este trabajo se usará el método de clasificación Haar Cascades (técnica de clasificación de cascada definida por Viola y Jones [2]) que es un enfoque clásico, ya que no hace uso de aprendizaje profundo (Deep Learning) al contrario de las CNN que sí lo hace (lo que implica más demanda de recursos computacionales). Además, este método que se aplica en este proyecto, es ideal para nuestros objetivos ya que vamos a trabajar con conjuntos de datos no muy grandes. En la Figura 1 se muestra a grandes rasgos el proceso que se sigue para la implementación del sistema referido.

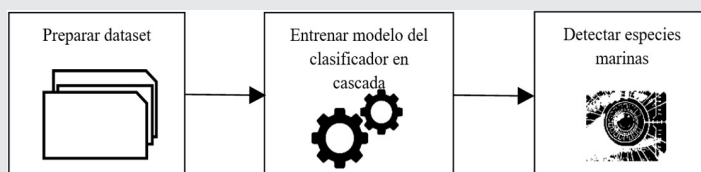


Figura 1. Pasos realizados para implementar el sistema de identificación.

Fuente: Elaboración propia.

Preparación de datos de entrenamiento

A continuación, se detallan los pasos que fueron aplicados para preparar los datos útiles en el entrenamiento de los algoritmos de reconocimiento de peces y tortugas.

1. Crear conjunto de datos

Para poder hacer el entrenamiento de nuestro clasificador es necesario tener tres conjuntos de datos, los cuales están conformados por imágenes, en un bloque se contiene información de los peces que se desean detectar (conjunto de muestras positivas) con un total de 50 elementos, otro bloque no contiene la

especie (peces) que se desea detectar (conjunto de muestras negativas) pero sí tiene un total de 50 registros de otro tipo de información. El último conjunto de datos contiene información de la especie que se desea detectar pero que no está en el conjunto de muestras positivas, este conjunto lo denominamos como las fotos de un video en formato “mp4” y/o imágenes en formato “jpg” (conjunto de prueba), tal y como se muestra en la Figura 2.

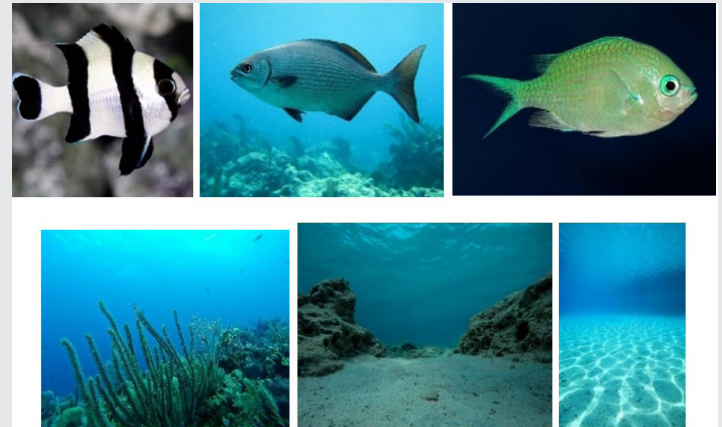


Figura 2. Ejemplo del conjunto de muestras positivas y negativas, respectivamente.

Fuente: Elaboración propia.

2. Preprocesar imágenes del conjunto de datos

Para proceder con nuestra construcción de un prototipo para reconocer peces y tortugas, en una primera etapa es importante considerar que las imágenes deben de ser del mismo tamaño contemplando además que sean lo más pequeñas posibles para que las etapas de entrenamiento y pruebas no impliquen muchos recursos computacionales, por lo tanto, se deben redimensionar; al final se obtienen imágenes de tamaño de 100 x 100. De igual manera, con el objetivo de facilitar el entrenamiento, se aplica a las imágenes escala de grises y con esto concluye el preprocesamiento de información. Véase la Figura 3.

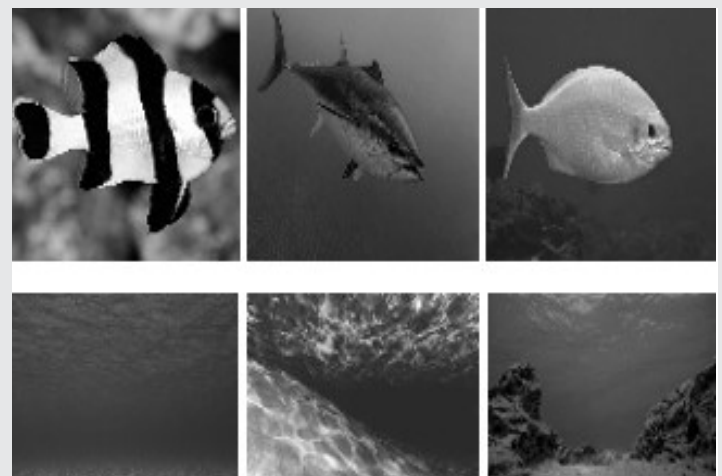


Figura 3. Ejemplo de imágenes preprocesadas.

Fuente: Elaboración propia.

3. Ampliación del tamaño de conjuntos de datos

Finalmente, para tener listo nuestro conjunto de datos, es necesario aumentar la cantidad de imágenes disponibles, entonces se procede a aplicar el efecto de espejo (cv2.flip) para obtener el doble de imágenes que se tenían previamente. Posteriormente, a estas imágenes se les aplicará rotaciones con diferentes ángulos (-60° , -40° , -20° , 20° , 40° , 60°) y obtenemos la cantidad de imágenes ampliada en un 600%, más las que teníamos. Tal y como se muestra en la Figura 4, para demostración se tiene el caso de ampliación de datos en el que por cada imagen se obtienen 13 adicionales.



Figura 4. Ejemplo de operaciones aplicadas a una imagen.
Fuente: Elaboración propia.

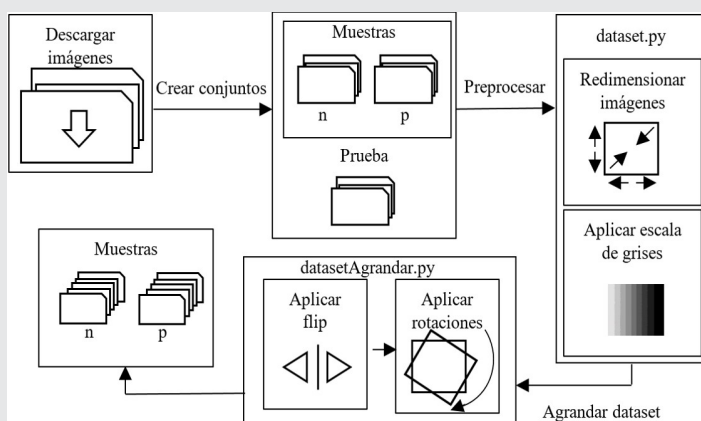


Figura 5. Pasos realizados para preparar los datos de entrenamiento.
Fuente: Elaboración propia.

En la Figura 5 se muestran gráficamente los pasos que se siguieron para conformar el conjunto de datos, listo para ser usado en el entrenamiento.

Entrenamiento de un clasificador en cascada

Para ejecutar el entrenamiento se utilizará la herramienta de software "Cascade Trainer GUI" y como se describió anteriormente, en este punto se debe contar con los conjuntos de imágenes muestra (negativo y positivo), teniendo en cuenta que el negativo debe ser mayor que el positivo, si es que el conjunto de datos es relativamente grande de lo contrario se puede tener el mismo volumen de datos, ya que el software antes mencionado se debe cargar con esta proporción de muestras o de otro modo podrá fallar el entrenamiento y mostrará un error aludiendo a un número insuficiente de muestras o algún otro tipo de error. Lo anterior se observó al estar realizando varias pruebas con diferentes conjuntos de datos de diferentes tamaños.

Primeramente, con el objetivo de seleccionar las muestras se indicará el porcentaje a tomar del conjunto muestra de imágenes positivas y el número de imágenes negativas que están en la muestra negativa, véase la Figura 6.

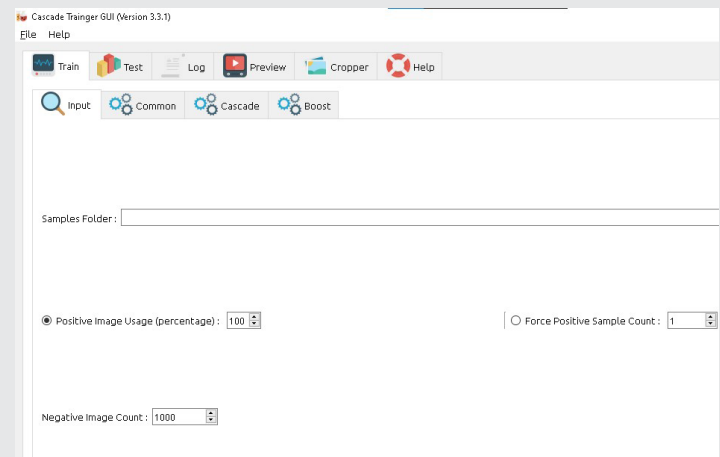


Figura 6. Interfaz para seleccionar muestras de entrenamiento.
Fuente: Elaboración propia.

Además, se determinan varios valores del número de etapas con el objetivo de obtener diferentes modelos como resultado del entrenamiento y así observar las diferencias, véase la Figura 7.

Posteriormente, se establecerá el ancho y la altura de la muestra por lo que es importante no configurar estos parámetros con tamaños muy grandes ya que el entrenamiento y la detección podrían ser procedimientos lentos. En este caso se configurará con 30X30, así mantenemos la relación de aspecto de nuestras imágenes puesto que en un principio en el preprocesamiento las dejamos con un tamaño de 100X100. Véase la Figura 8.

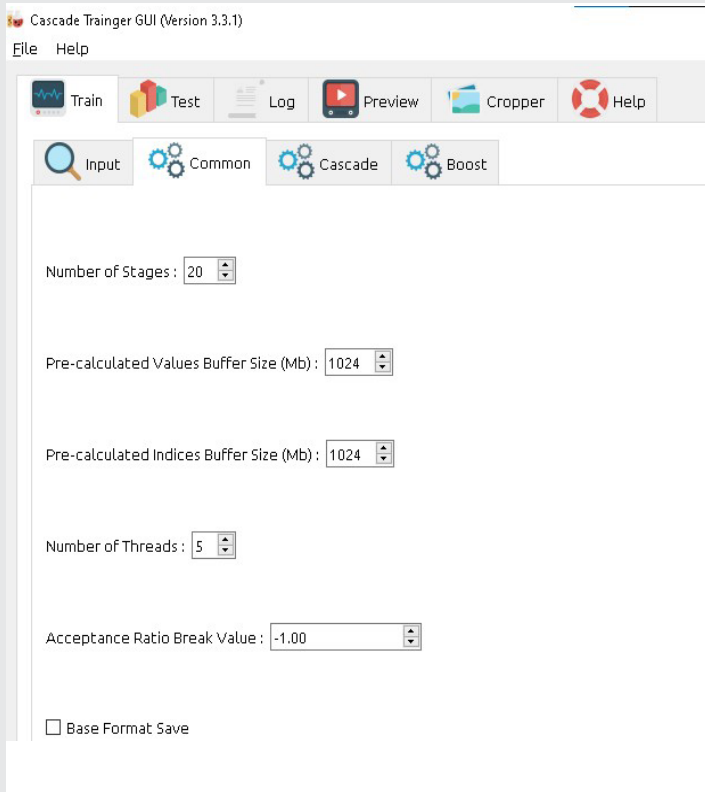


Figura 7. Parámetros de entrenamiento.
Fuente: Elaboración propia.

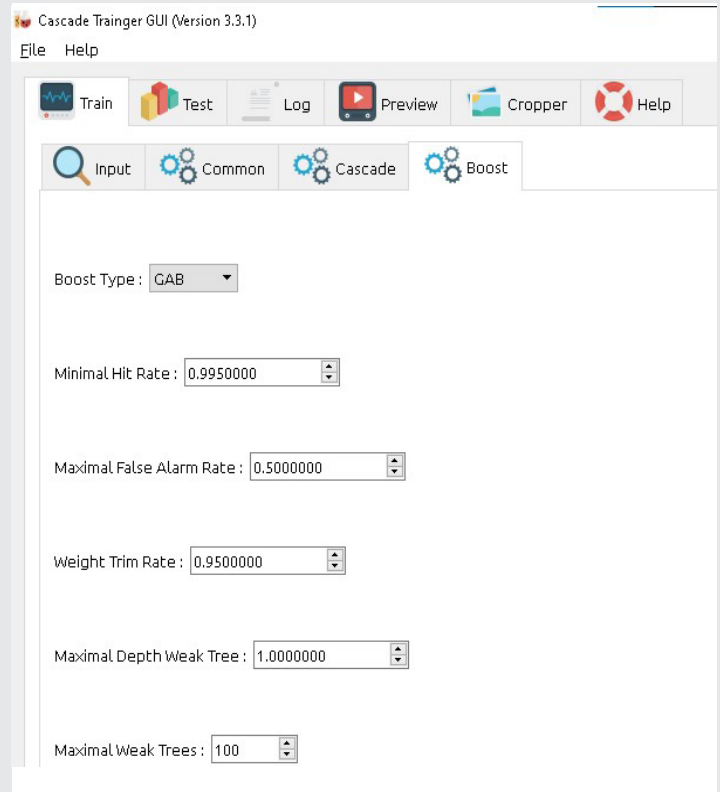


Figura 9. Ejecución del entrenamiento.
Fuente: Elaboración propia.

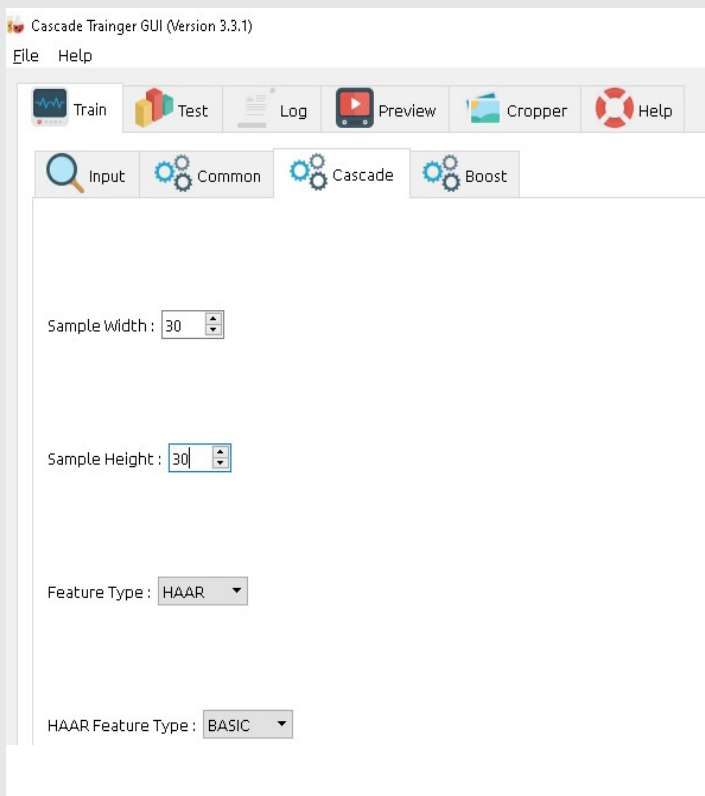


Figura 8. Configuración de otros parámetros para el entrenamiento. Fuente: Elaboración propia.

Al terminar el entrenamiento, hemos construido un modelo a partir de los datos previamente referidos. Este modelo se usará para ejecutar pruebas con datos no vistos por el algoritmo en la etapa de entrenamiento. Véase en la Figura 10 los pasos realizados para el entrenamiento del clasificador en cascada.

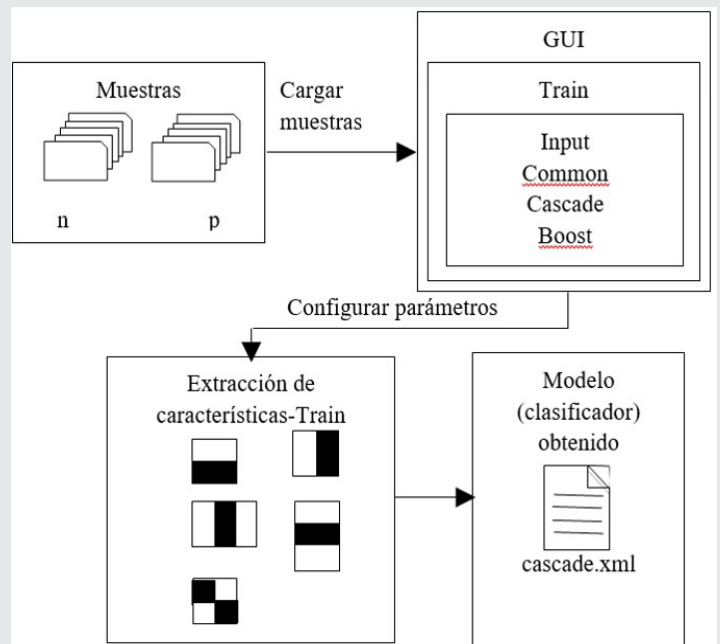


Figura 10. Pasos realizados para el entrenamiento del clasificador en cascada.
Fuente: Elaboración propia.

Finalmente, se procederá a la ejecución del entrenamiento del modelo, véase la Figura 9.

Detectar especies marinas

Una vez obtenido el modelo del entrenamiento del clasificador en cascada (cascade.xml) podemos poner a prueba el detector. Como muestra de prueba se usa un video (conjunto de fotografías) en formato "mp4", una vez ejecutada la detección se va a mostrar como salida al mismo video en donde, si se detecta un pez, se pintará un marco verde sobre este. En la Figura 11 se observa esquemáticamente el proceso de detección.

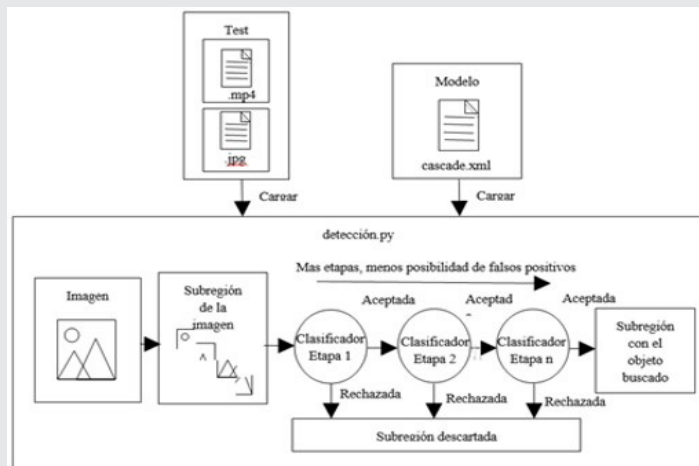


Figura 11. Proceso de detección.

Fuente: Elaboración propia.

Entre mejor haya sido entrenado el clasificador, habrá menos posibilidad de falsas identificaciones.

RESULTADOS

Ya que tenemos claro el proceso para implementar un sistema de identificación, desde la preparación del conjunto de datos, el entrenamiento del modelo del clasificador en cascada, hasta la identificación de la especie marina deseada; entonces podemos entrenar un clasificador para tortugas y mejorar la eficiencia del clasificador para peces. Las mejoras con las que vamos a experimentar con cada clasificador referido se muestran en la Tabla 1.

Tabla 1. Configuración de clasificadores.

experimento / clasificador	Peces	Tortugas
Modificar número de etapas en el entrenamiento	X	X
Cambiar cantidad de las muestras	X	X
Modificar parámetros del módulo detectMultiScale	X	X
Cambiar muestras positivas por una característica en específico (caparazón, vista superior)		X
Ejecución de diferentes clasificadores (peces y tortugas) en conjunto (detección múltiple)	X	X
Crear y ejecutar por separado clasificadores de diferentes vistas de una característica específica		X
Ejecución en conjunto de clasificadores de diferentes vistas de una característica específica		X

Fuente: Elaboración propia

Número de etapas en el entrenamiento

En estas pruebas de nuestro trabajo, se entrenaron dos modelos diferentes (uno con 10 etapas y otro con 20) por cada especie (pez y tortuga), con el propósito de ver las diferencias en las detecciones. Sin embargo, también se requiere la construcción de un código diferente para cada clasificador, ya que se configuran de diferente forma por tratarse de diferentes objetos. Los modelos clasificadores que ofrecieron mejores resultados en esta investigación fueron los que se entrenaron con 10 etapas. En la Figura 12 se observa la diferencia, a pesar de tratarse de una única fotografía la que se muestra, a lo largo de los videos siempre se comportan mejor los clasificadores de 10 etapas.

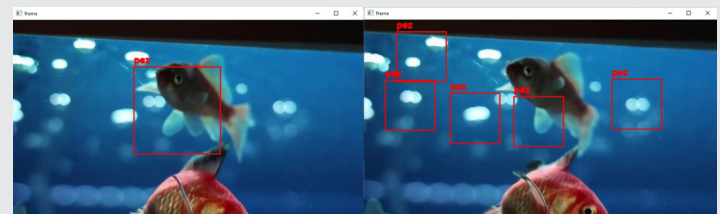


Figura 12. Ejecución del clasificador entrenado con 10 y con 20 etapas, respectivamente.

Fuente: Elaboración propia.

En la Figura 13, véase la diferencia entre una óptima y una mala configuración de un clasificador.

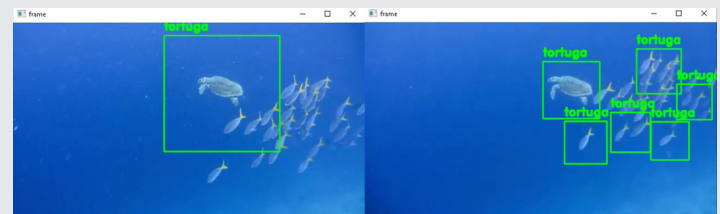


Figura 13. Diferencia entre una óptima y una mala configuración de un clasificador, respectivamente.

Fuente: Elaboración propia.

Muestras positivas representando una característica específica

A partir de nuestros diversos experimentos, se decidió centrarnos en una característica para hacer la detección de tortugas, podía ser el rostro o el caparazón, se optó por el caparazón desde una vista superior. En este sentido, en el proceso de creación del conjunto de datos positivo se agregaron varias muestras de estos caparazones. En este entrenamiento se usaron 110 imágenes para la muestra positiva y 230 para la muestra negativa. En la primera versión se observa cómo el clasificador es capaz de identificar tortugas en su totalidad y en la mayoría de sus posibles vistas, sin embargo, también generaba más identificaciones falsas. Además, esta versión del clasificador no puede identificar tortugas que se encuentran sin contraste con el fondo y con una vista superior, ya que las muestras para entrenar este clasificador son más generales y trata de identificar aletas y cabeza. En la segunda versión podemos observar que no puede identificar aquellas tortugas en las que el ángulo de su caparazón

no se muestra en su totalidad, pero es muy preciso al identificar caparazones que la primera versión era incapaz de identificar. Véase la Figura 14.

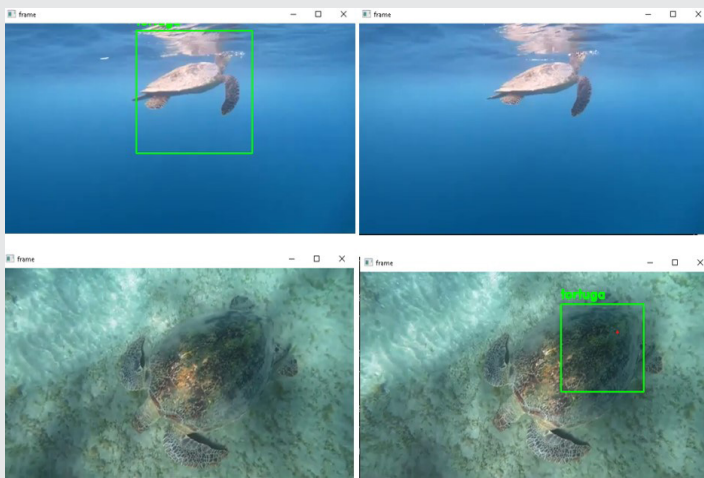


Figura 14. Diferencia entre dos versiones del clasificador. Izquierda.: versión 1; Derecha.: versión 2.
Fuente: Elaboración propia.

Ejecución de diferentes clasificadores (peces y tortugas) en conjunto (detección múltiple)

En esta etapa de prueba, hacemos múltiples detecciones, cargamos el clasificador de tortugas y el clasificador de peces en un mismo programa, por lo tanto, esperamos que el sistema detecte peces y tortugas en una misma ejecución. Se tuvieron algunas dificultades ya que el clasificador de peces hacía falsas identificaciones y clasificaba como pez a la aleta de las tortugas; mientras que el modelo clasificador de tortugas funcionó de manera correcta en sus dos versiones, cada una con sus respectivos detalles que se mencionaron anteriormente, pero en general la eficiencia de este clasificador es satisfactoria. Véase la Figura 15.

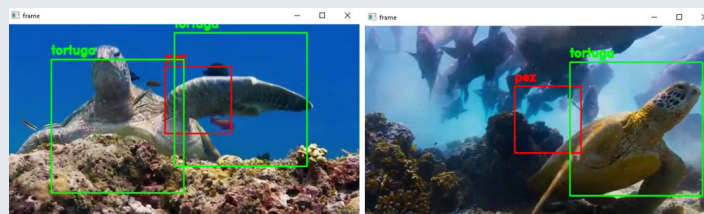


Figura 15. Ejemplo de multi detección.
Fuente: Elaboración propia.

Clasificadores de diferentes vistas de una característica específica

En esta prueba se repitió cuatro veces el procedimiento para generar el sistema de identificación de especies marinas (que se ha explicado previamente) ya que queremos obtener cuatro de seis vistas posibles de una característica en específico de la tortuga. La característica que se va a usar en esta prueba es la cabeza de la tortuga y de esta queremos los clasificadores de la vista lateral izquierda, lateral derecha, frontal y superior. Aunque un simple fragmento de un

video es poca evidencia del funcionamiento de los clasificadores, los de las vistas laterales tuvieron una buena eficiencia. Con respecto al clasificador frontal de la cabeza de la tortuga, fue muy baja su eficiencia, se puede decir que el ángulo en los fragmentos de video no favoreció mucho para esta tarea puesto que las tortugas nunca estuvieron completamente de frente. De igual forma el clasificador de vista superior tuvo muchas deficiencias, hacía falsas identificaciones y a menudo tomaba las aletas o el caparazón de las tortugas como una cara. Véase la Figura 16.

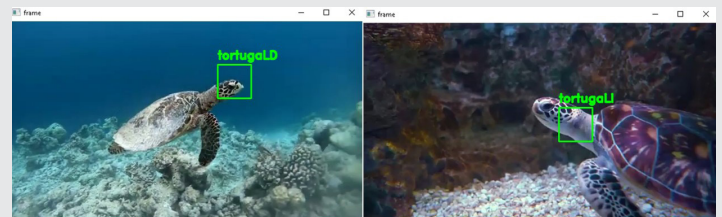


Figura 16. Identificación de la cara lateral, derecha e izquierda de una tortuga.
Fuente: Elaboración propia.

Ejecución en conjunto de clasificadores de diferentes vistas de una característica específica

En este ejercicio trabajamos con los clasificadores de las diferentes vistas de la cabeza de las tortugas que tuvieron una mejor eficiencia, estos fueron los clasificadores de las vistas laterales, tanto izquierda como derecha. Para saber cuál clasificador identificaba una vista, se diferenciaron por colores, tal y como se muestra en la Figura 17.

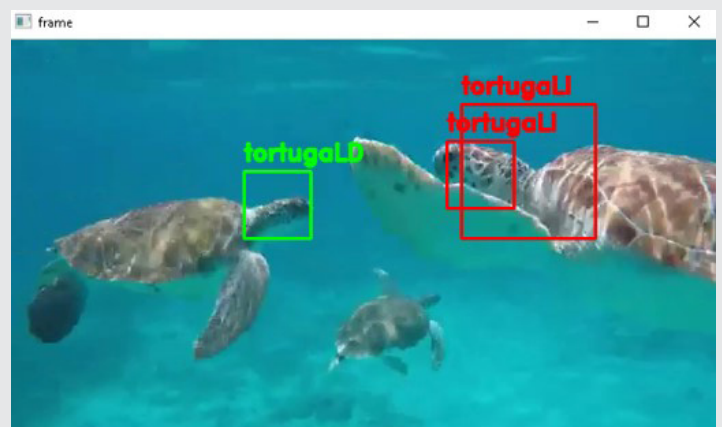


Figura 17. Identificación de caras de tortugas en diferentes vistas.
Fuente: Elaboración propia.

Anteriormente se había experimentado con el caparazón (una característica) de la tortuga para hacer la identificación correspondiente, lo cual resultó en una muy buena eficiencia. Hay que tomar en cuenta que el caparazón es simétrico, por lo tanto, no hubiera tenido mucho caso hacer diferentes clasificadores para sus diferentes vistas, podemos decir que con la vista superior era más que suficiente para que el sistema identificador encontrara el patrón del caparazón en las muestras de prueba.

Catalogamos al caparazón como la mejor característica en la tortuga para intentar identificarla, puesto que, al estar hablando de objetos en movimiento, en cualquier momento va a mostrarse esta característica de la especie; al contrario de la cabeza de la tortuga que es una característica que no se hace notar mucho y es más difícil de identificar, ya que interfiere mucho la calidad de las muestras y demás factores.

En la Figura 18 se muestran algunas identificaciones de especies marinas.

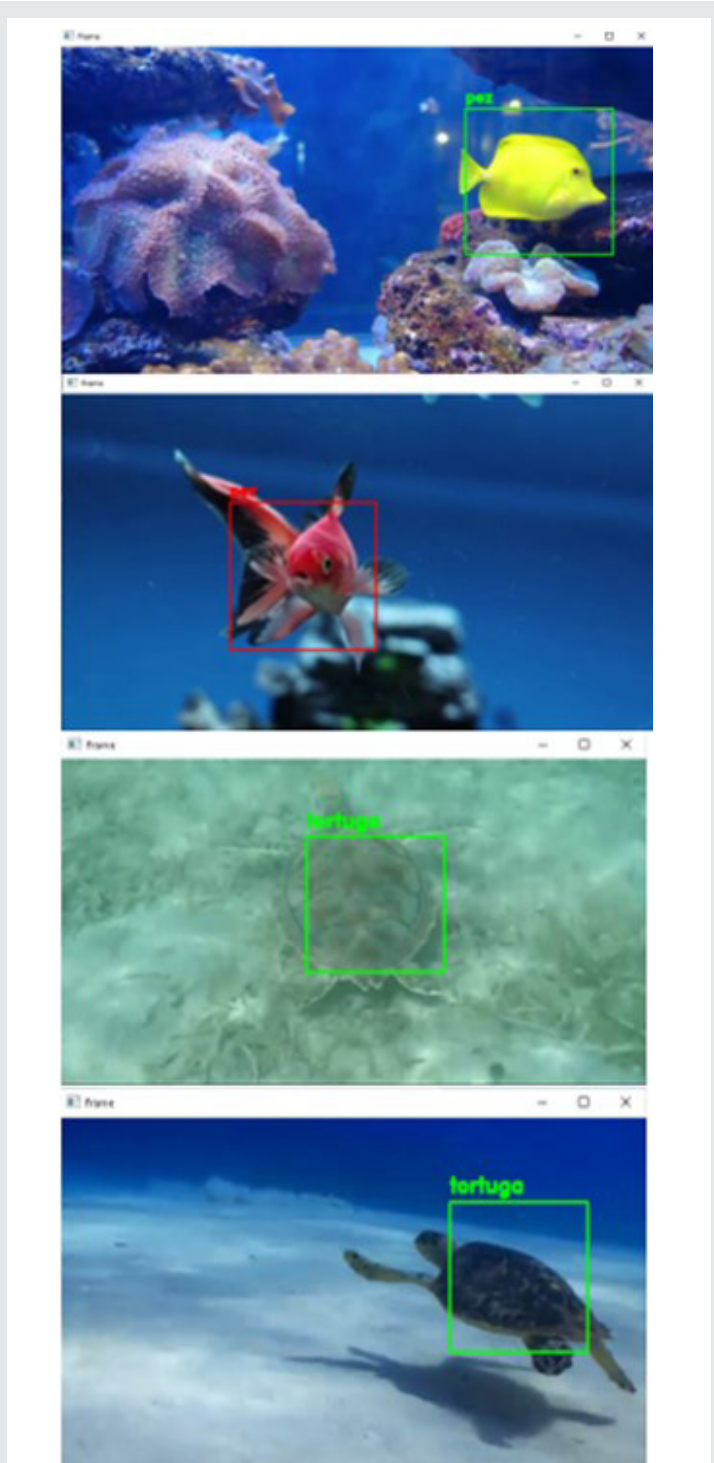


Figura 18. Identificación de especies marinas.

Fuente: Elaboración propia.

CONCLUSIONES

Durante el desarrollo de este proyecto se comprendieron los conceptos y las diferencias entre los métodos que se pudieron construir para el entrenamiento del modelo aplicado en la detección, las ventajas y desventajas de cada uno, para que finalmente se optara por el que más se adaptara a nuestros requerimientos.

Las dificultades a las que nos enfrentamos fueron varias, desde la usabilidad del software utilizado para el entrenamiento del clasificador en cascada, ya que no podíamos finalizar el entrenamiento con éxito para construir el clasificador. Se experimentó con diversas variables tales como la cantidad de muestras positivas y negativas que se ponían a disposición; también se experimentó con los parámetros que este software ofrecía principalmente con las etapas, de esta manera pudimos analizar los diferentes resultados (detecciones).

La principal dificultad fue la aleatoriedad de ambientes (fondo o escenario) en las muestras, de tal forma que influye negativamente en las detecciones. Al contar con muestras de un ambiente no controlado como lo es el fondo marino van a variar mucho las características de las imágenes, como el tamaño, la resolución, la distancia focal, las sombras, la luminosidad, la perspectiva, etc. Esto ocasiona que baje la eficiencia de nuestro sistema.

A pesar de todo pudimos obtener clasificadores funcionales tanto de peces como tortugas en diferentes versiones, por cada uno de estos se pudo apreciar las ventajas y desventajas, de tal forma que usamos en pruebas posteriores los que brindaban mayor eficiencia. Conforme se avanzaba en las pruebas pudimos ser más específicos con los clasificadores; pasamos de identificar una especie en general, a una característica importante de una especie, hasta identificar una vista de una característica importante de una especie. Se partió de lo general a lo particular para tener un sistema más eficiente. El objetivo de centrarnos en una característica importante de una especie es porque se intenta identificar objetos en movimiento y en cualquier momento se logrará con esa característica destacada de la especie; de tal forma que el sistema dispondrá de una precisión de entre el 70% y 80%.

Podemos observar como la inteligencia artificial cada vez más va tomando un papel importante en el ámbito ambiental, con capacidad de predecir y detectar; nuestro sistema no es la excepción y en un futuro veremos más sistemas como este implementados para la solución de problemáticas presentes en la vida real.

BIBLIOGRAFÍA

[1] Organización de las Naciones Unidas Desarrollo Sostenible. (n.d.). Océanos - Desarrollo Sostenible.

Obtenida el 1 Julio de 2022, de la página electrónica: <https://www.un.org/sustainabledevelopment/es/oceans/>.

[2] Viola, P.; Jones, M. (2001). *Rapid Object Detection using a Boosted Cascade of Simple Features*. *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 511–518.

[3] Jeremías, E.A. (2020). *Reconocimiento de objetos a través de la metodología Haar Cascades*. En *RADI. Argentina*. Vol. 16.

[4] OpenCV. (2020). *Cascade Classifier Training*. Obtenida el 10 de Julio de 2022, de la página electrónica: https://docs.opencv.org/4.3.0/dc/d88/tutorial_traincascade.html.