

# Sistema de detección y clasificación de peces utilizando visión computacional

Rodrigo Mejía  
20140809@aloe.ulima.edu.pe / Universidad de Lima, Perú

Gianfranco Rosales  
20141199@aloe.ulima.edu.pe / Universidad de Lima, Perú

Recepción: 21-6-2019 / Aceptación: 8-8-2019

**RESUMEN.** La gestión de los recursos hidrobiológicos implica tanto el aspecto ecológico a través del equilibrio del ecosistema, como el aspecto económico mediante el control de la cantidad y calidad de los recursos pesqueros producidos en el Perú. En la actualidad, labores relacionadas a esta gestión son realizadas por empresas privadas y entidades del Estado como el Imarpe. La misión de estas es proteger la calidad de los recursos que llegan a los hogares de millones de peruanos. Esta investigación busca desarrollar un sistema para la detección, clasificación y, finalmente, la medición de diversas especies de peces, utilizando técnicas de visión computacional como el algoritmo SURF y redes neuronales convolucionales. Las pruebas, utilizando dos especies de peces, demostraron que la identificación alcanza un nivel de precisión del 90 % y que la clasificación alcanza una precisión del 80 %. Estos valores se obtienen bajo determinadas condiciones que se comentan en el desarrollo del artículo.

**PALABRAS CLAVE:** visión computacional, SURF, *background subtraction*, *border detection*, red neuronal convolucional, *machine learning*, detección, clasificación, inceptionV3, peces, CNN

## Fish Detection and Classification System Using Computational Vision

**ABSTRACT.** The management of hydrobiological resources involves both the ecological aspect through the balance of the ecosystem, and the economic aspect through the control of the quantity and quality of the fishery resources produced in our country. Currently, work related to this management is carried out by private companies and state entities such as Imarpe. Their mission is to protect the quality of the resources that reach the homes of millions of Peruvians. This research aims to develop a system for the detection, classification and finally measurement of various species of fish, using computational vision techniques such as the SURF algorithm and convolutional neural networks. The tests, which used two fish species, showed that the identification reaches a 90% accuracy level and the classification reaches an 80% accuracy level. These values are achieved under certain conditions that are discussed in the article.

**KEYWORDS:** computational vision, SURF, background subtraction, border detection, convolutional neural network, machine learning, detection, classification, inception V3, fish, CNN

## 1. INTRODUCCIÓN

Los recursos marinos representan una fuente de recursos valiosa para el desarrollo de la economía y la alimentación de sus comunidades tanto para el mundo como para nuestro país. La Organización de las Naciones Unidas para la Alimentación y la Agricultura (FAO, por sus siglas en inglés), indica que el consumo aparente de kilogramos per cápita para la alimentación y otros fines es cercano a la utilización total de recursos disponibles. Particularmente, nuestro país es uno de los países más ricos en recursos pesqueros del planeta, ocupando el quinto lugar de producción pesquera en el año 2016 (FAO, 2018). Durante el período enero-junio del 2018, el Perú exportó 839,3 toneladas de pescado, lo cual en términos monetarios significó 1787,6 millones de dólares (Sociedad Nacional de Pesquería, 2018).

Acorde al Codex Alimentarius de la FAO (CRFM, 2016), la trazabilidad es definida como “la capacidad de seguir el movimiento de un alimento a través de etapas específicas de producción, elaboración y distribución”. En el caso de instituciones como el IMARPE (Instituto del Mar del Perú), en donde realizan la clasificación y medida de especies de forma manual, le podría generar un ahorro de esfuerzo y tiempo automatizando estos procedimientos. En el caso de terminales pesqueros también podrían acceder a este ahorro, facilitando la trazabilidad debido al menor tiempo para analizar los peces.

Dichas estadísticas y antecedentes expresan la importancia de contar con recursos de calidad que, al ser extraídos del mar, presenten condiciones óptimas para la llegada al consumidor final. Por ello, en esta investigación se plantea un sistema de detección, clasificación y medida de determinadas especies de peces, tarea realizada manualmente en los laboratorios de instituciones como el Instituto del Mar del Perú y que, de desarrollarse adecuadamente, significaría un ahorro de esfuerzos importantes para este tipo de organizaciones.

## 2. ESTADO DEL ARTE

En esta sección se analizan trabajos relacionados a las aristas claves de esta investigación: la detección de objetos y la clasificación de imágenes mediante aprendizaje supervisado. Por un lado, la discusión sobre detección de objetos abarca trabajos relacionados en otros campos de estudio (p. ej., detección de ballenas) y la comparación de resultados del algoritmo SURF con otros afines como SIFT, ORB, entre otros. Acto seguido, para el análisis de los trabajos relacionados a la tarea de clasificación de imágenes, se discute la aplicación de redes neuronales convolucionales (también en su abreviatura como CNN) en la tarea de clasificación, así como las arquitecturas de las mismas, y aplicaciones e investigaciones orientadas al reconocimiento de peces.

## 2.1 Detección de objetos en otros campos de estudio

Las investigaciones relacionadas a la detección de objetos específicos vinculados a la visión computacional son las siguientes:

Nadarajan, Chen-Burger y Fisher (2011) diseñaron un sistema capaz de analizar vídeos bajo el agua para detectar, rastrear y contar peces. Para la detección de peces, los autores utilizaron el modelo gaussiano y el H-model que ahorra el costo computacional, donde los valores de precisión terminaron entre 80,5 %–89,5 %.

Mahajan y Perkins (2015) se enfocaron en detectar e identificar ballenas en un conjunto de imágenes tomadas de manera aérea. Para la detección de ballenas se utilizaron las técnicas *cascade object detector* (COD) y el método de agrupamiento de K-medias (KMC). Para la identificación de las ballenas utilizaron SIFT y *support vector machine* (SVM). Los resultados de esta investigación concluyeron que KMC detecta mejor que COD en el 90 % de los casos, mientras que no consiguieron que los métodos de identificación funcionen de manera satisfactoria.

Chincha y Tian (2011) propusieron un método de reconocimiento de objetos para ayudar a invidentes para que encuentren sus pertenencias mediante el algoritmo de visión computacional SURF. Los objetos por detectar eran llaves, teléfonos celulares, billeteras, lentes de sol y zapatillas. Este método propuesto por los autores terminó con un 84 % de precisión para los cinco tipos de objetos. Por otro lado, Shujuan, Zhize, Xingang, Guan, Wenqi y De (2015) utilizaron el mismo algoritmo para realizar una detección de vehículos a tiempo real. Sus resultados fueron positivos, solo con pequeños percances como la confusión entre el carril y el vehículo, y en condiciones de alta y baja luminosidad, donde el detector de vehículos no funcionaba de la manera esperada.

## 2.2 Reconocimiento de peces

Chen, Sun y Shang (2017) proponen un sistema de clasificación de peces automático para el etiquetado de los peces usando tecnologías de aprendizaje profundo, específicamente modelos combinados de tecnologías de redes neuronales convolucionales. La clasificación se trabajó con distintas arquitecturas combinadas como ResNet50, InceptionV3 y VGG16. La técnica que obtuvo un mejor resultado respecto a la función de error fue la técnica InceptionV3, entrenada con *data* externa con el valor de 0,853 de pérdida.

Frost y Geisler (2016) comparan los métodos de clasificación bayesianos ingenuos, máquinas de vectores de soporte y árboles aleatorios para clasificar especies de peces del mismo conjunto de datos trabajado por Chen, Sun y Shang (2017). La precisión de clasificación obtenida es de 7 %, 40 % y 80 % para cada método de clasificación, respectivamente. Posterior a la optimización del método de árboles aleatorios se logró obtener un valor de precisión general del modelo en el entrenamiento de 99,5 % y en la fase de prueba de 83,4 %.

Dada la revisión de literatura mencionada, el equipo de trabajo concluyó que las técnicas de redes neuronales convolucionales profundas y el algoritmo SURF son válidas para la propuesta de solución.

### 3. METODOLOGÍA

Esta sección detallará determinados tópicos para el entendimiento del nacimiento de la solución que se plantea. Primero, hace referencia a las etapas de la investigación. A continuación, hace referencia y detalla la estructura de la solución propuesta. Luego, se mencionan los métodos que se probaron durante el proceso. Finalmente, se explica la manera en que se validó el trabajo realizado.

#### 3.1 Etapas de desarrollo de la solución

Durante la primera fase, se buscaron métodos para resolver el problema planteado: poder detectar y clasificar peces en imágenes que presenten determinada escena. Para lograr esto, se investigó arduamente de manera que se identificaron métodos potenciales para el planteamiento de una futura solución: emplear algoritmos *hand-crafted* como lo son SURE, SIFT, etc., y el uso de redes neuronales convolucionales profundas para la tarea de clasificación de imágenes.

La siguiente fase o etapa consistió en la obtención de los datos (imágenes) a trabajar. Para el desarrollo de la prueba de concepto que se realizó, se empleó un conjunto de datos obtenidos de la organización Kaggle. Gracias a los optimistas resultados obtenidos, se propuso mejorar y ajustar la solución preliminar propuesta y trabajar con datos reales, imágenes que presenten a los peces que se encuentran en nuestro litoral.

Esto da pase a la tercera etapa: la obtención de los datos finales. Para esta labor, el equipo de trabajo se contactó con múltiples organizaciones. En última instancia, se recibió el apoyo del Instituto del Mar del Perú. Gracias al apoyo del ingeniero Julio Alarcón, jefe de Artes de Pesca de la institución, el equipo pudo programar una visita guiada a cargo del personal de Gestión de la Calidad del Terminal Pesquero de Ventanilla, a fin de capturar las imágenes que se emplean en el presente trabajo. Finalmente, como última etapa se encuentra la optimización de los métodos. Esta se realizó en base a los resultados empíricos obtenidos durante la fase de experimentación.

#### 3.2 Estructura de la solución

La solución propuesta posee determinados procesos que el sistema realizó para conseguir el objetivo planteado, donde se empezó con el preprocesamiento de las imágenes y se culminó

con la visualización e interpretación de los resultados de clasificación en la interfaz gráfica de usuario.

Se realiza el preprocesamiento de imágenes. Este consiste en dos tareas esenciales: el redimensionamiento de las imágenes y el entrenamiento realizado de manera iterativa, evaluando y analizando los resultados obtenidos de los valores de pérdida y precisión que se obtienen durante el proceso de entrenamiento y validación. La salida de este último mencionado es un archivo en formato HDF5 conteniendo los pesos y conexiones ideales generadas en el proceso de entrenamiento de la red. Este proceso fue realizado íntegramente en la nube de Google Colaboratory, por lo que se logró cruzar la barrera de recursos computacionales que se tuvo al comienzo de la investigación.

Paralelamente, se realizó la detección de características de una imagen base, donde el algoritmo SURF trabajó con una silueta característica de un pez. Posteriormente, se extrajeron las características de las imágenes reales, donde se quiso identificar y detectar al pez en un contexto real. Este último paso se logró gracias al uso de la técnica *background subtraction* y *border detection*, donde el primero se encarga de abstraer al pez del contexto total de la imagen y el segundo de detectar los bordes de la imagen. Tras la detección, se hizo el *feature matching* entre ambas imágenes, donde se analiza si la imagen del conjunto de datos tiene las características de la imagen base. Si la imagen las contiene, se procesa el pez detectado generando una imagen, la cual sirve de entrada a la red neuronal convolucional para poder realizar la clasificación en las imágenes. Finalmente, teniendo como entrada los pesos generados en el entrenamiento y la imagen del pez detectado, se clasifica. En la siguiente figura se muestra de manera gráfica el proceso descrito.

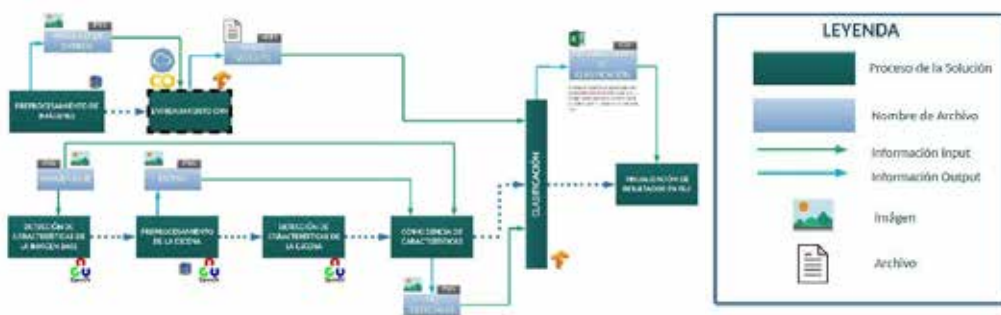


Figura 1. Estructura de la solución propuesta

Elaboración propia

### 3.3 Métodos probados

Respecto a la tarea de clasificación, se probaron tres arquitecturas de red neuronal convolucional conocidas. Estas fueron VGG-16, ResNet e InceptionV3. Se optó, finalmente, trabajar con la última mencionada por presentar los mejores resultados preliminares en base a las pruebas realizadas. Por ello, en esta investigación este modelo se optimiza a nivel de parámetros y agregación de capas con el fin de obtener las mejores conexiones y pesos para el modelo final. En la siguiente figura se observa gráficamente el modelo InceptionV3 con la agregación de capas en la parte superior. En la parte inferior se observan las últimas capas del modelo original de InceptionV3, las cuales son precargadas con los pesos de ImageNet. Por otro lado, las capas agregadas se entrenan con las imágenes del conjunto de imágenes de peces obtenidos por el equipo de trabajo.

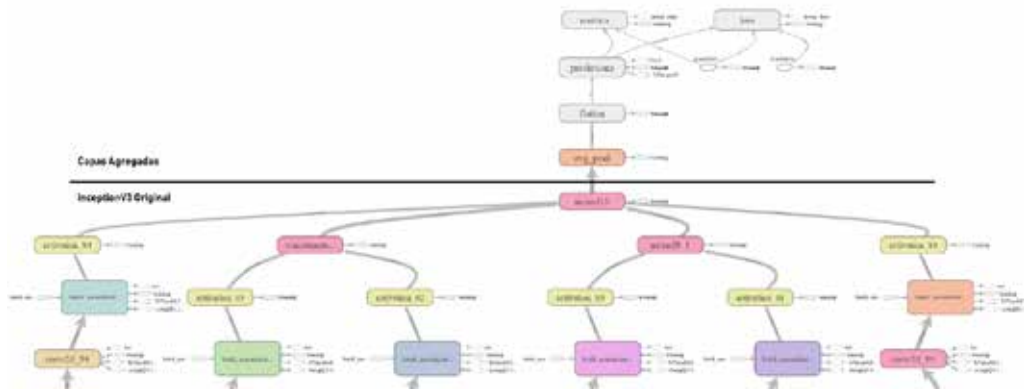


Figura 2. Modelo InceptionV3 adaptado a la propuesta  
Elaboración propia

Por otro lado, para la tarea de detección se probaron tres algoritmos de visión computacional: SIFT, SURF y ORB. Se optó trabajar con SURF porque detectó satisfactoriamente a los peces en sus respectivas escenas. Otras técnicas implementadas para poder facilitar la detección fueron *background subtraction* y *border detection*.

## 4. EXPERIMENTACIÓN

Se decidió utilizar un arco para poder realizar la toma de las fotos con los siguientes equipos computacionales: una cámara Logitech c922 Pro Stream que tiene como resolución máxima 1080p/30fps y un ángulo de visión de 78°, un ordenador de procesamiento remoto de Google Colaboratory (GPU Tesla K80 con 2496 CUDA cores) y un ordenador con las librerías instaladas.

#### 4.1 Detección del pez en la escena




Para la detección de los peces se utilizaron las siguientes técnicas: el algoritmo de visión computacional SURF, la técnica de *background subtraction* y *border detection*. Para la detección y caracterización de la imagen base, fue utilizado un contorno de un pez, el cual presenta las siguientes condiciones:

- Fondo blanco.
- Es únicamente un contorno genérico de un pez.
- La imagen tiene características que todos los peces del conjunto de datos poseen.

Para empezar, se procesó la imagen base, encontrando sus características mediante el algoritmo SURF. Posteriormente, se aplicó *background subtraction* en la escena del pez que se quiere detectar su presencia en la escena, donde se comparan dos imágenes: la de la escena sin el pez y la del pez en la escena. En las imágenes presentadas, se muestra el pez cojinova.

Tabla 1

*Imágenes a procesar mediante background subtraction (fondo y con la escena) y su posterior resultado*

Imagen			
Descripción	Imagen “fondo” sin el pez en la escena	Imagen con el pez en la escena	Resultado tras el <i>background subtraction</i>

Elaboración propia

Como se puede observar en la tabla 1, las imágenes tras el *background subtraction*, el pez se ve de manera clara. Posteriormente, se aplica el *border detection* para poder detectar el contorno de estas imágenes.



Figura 3. Resultado tras aplicar *border detection* en el resultado anterior

Elaboración propia

Debido a que se aplicó el *background subtraction*, previamente, el contorno del resto de peces en la imagen no se visualiza de manera tan clara como el del pez objetivo. Ya teniendo el borde detectado de ambas imágenes (la imagen a detectar y la silueta genérica) se procede a la etapa de coincidencia de características detectadas entre ambas imágenes.

Esta etapa de coincidencia de características (*feature matching*, en inglés) se realiza mediante FLANN (Fast Library Approximate Nearest Neighbor), la cual es una librería que permite comparar los puntos de características encontrados en ambas imágenes. Tras realizar la coincidencia de características, se pasa a la prueba de ratios de distancia entre los puntos vecinos para filtrar la similitud de coincidencias, donde se definen las coincidencias que van a ser tomadas en cuenta.

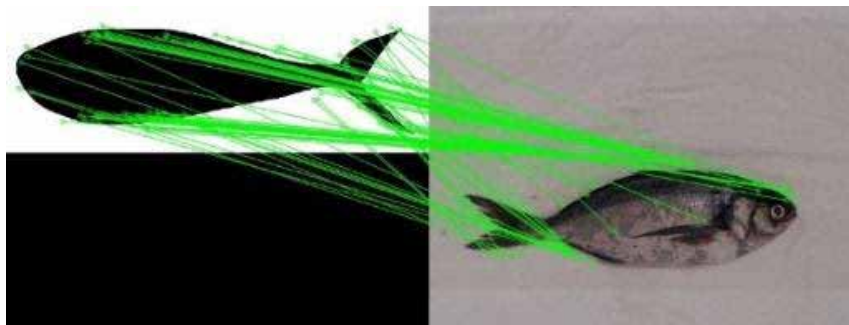


Figura 4. Resultados de la detección

Elaboración propia

Finalmente, la parte de detección del pez en la escena concluye devolviendo como resultado el pez detectado y extraído de la escena para su posterior clasificación.





Figura 5. Exhibición del *output* final de la etapa de detección

Elaboración propia

## 4.2 Clasificación de peces

Para la ejecución de los experimentos y pruebas relacionadas a la tarea de clasificación, se realizaron determinados pasos para lograr el entrenamiento exitoso en la nube de Google Colaboratory.

Siendo el objetivo subir las imágenes a la carpeta de trabajo en la nube de Colaboratory, primero se hicieron determinadas tareas de forma local. Las tareas realizadas se detallan en el siguiente orden:

- 1) Etiquetado de las imágenes para poder saber qué imagen corresponde a determinada especie. Esta tarea se realizó de forma manual.
- 2) Para las imágenes que contaban con el formato HEIC, se empleó un *software* de conversión de formato a JPEG.
- 3) División del conjunto de datos en una proporción de 80 % para entrenamiento y 20 % para validación.
- 4) Dadas las dimensiones originales de las imágenes: 4032 x 3024 píxeles, se redimensionaron las mismas con el objetivo de optimizar el tiempo del proceso de entrenamiento en la nube. Las nuevas dimensiones obtenidas fueron de 299 x 299 píxeles.

### 4.3 Entrenamiento de la red neuronal

Se realizaron numerosas iteraciones de entrenamiento caracterizadas por la variación de los parámetros de tamaño de lote, cantidad de épocas y el parámetro de guardado del modelo y sus pesos. El mejor resultado obtenido se dio en la iteración que presentó mejor convergencia en el tiempo de los valores de precisión y pérdida de entrenamiento y validación durante el proceso de entrenamiento. Los valores de los parámetros para la iteración mencionada son: tamaño de lote de 32, número de épocas 30 y ratio de aprendizaje de 0,001.

Los valores de precisión obtenidos durante esta fase son de 0,977 para el entrenamiento y 0,962 para la validación. Asimismo, los valores de pérdida y entrenamiento obtenidos son de 0,18, aproximadamente. Estos se pueden apreciar en la figura 6.

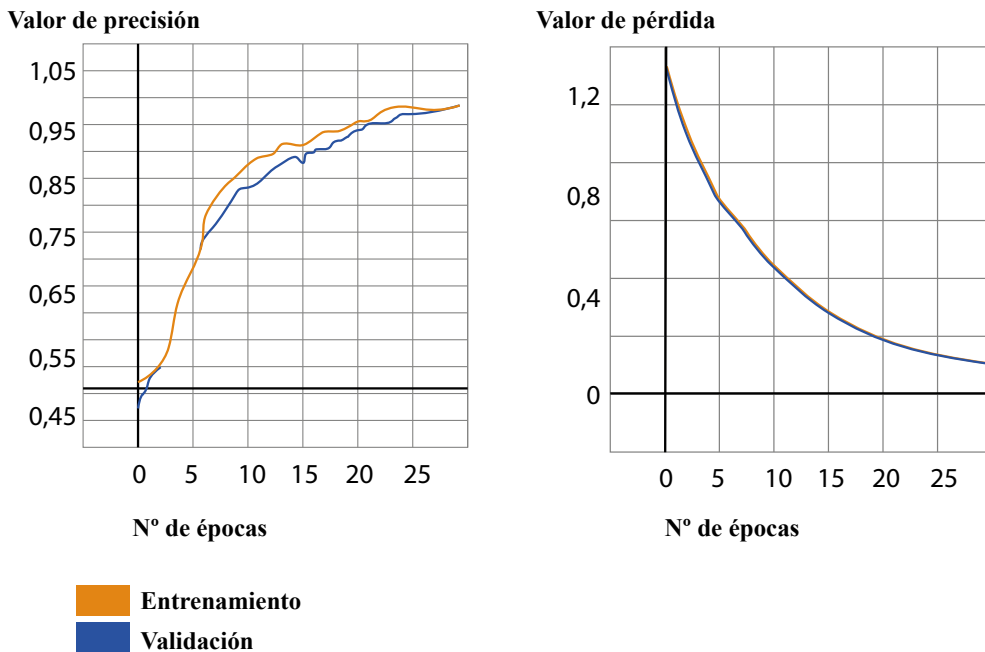


Figura 6. Valores de precisión y pérdida de entrenamiento y validación

Elaboración propia

## 5. RESULTADOS

Los hallazgos de este trabajo de investigación se presentan en dos etapas de experimentación: detección y clasificación; y en dos fases, la fase de pruebas con imágenes impresas de los peces y las pruebas realizadas con peces reales.

En la primera fase se obtuvieron los siguientes resultados:

Respecto a la detección, se procedió a detectar un conjunto de imágenes que tenía las siguientes características: un total de veinte imágenes, siete imágenes por cada especie y seis imágenes sin un pez en la escena. Se obtuvo lo siguiente:

- Tras la implementación de *background subtraction*, *border detection* y la conversión de la imagen a binaria (*thresholding*), el algoritmo detectó satisfactoriamente a los peces en la escena.
- Se detectó correctamente el 90 % de las imágenes.
- El recorte de la imagen, tras agregar ciertos valores de tolerancia (p. ej., si se detectó la cola de un pez de manera parcial, se recorta con un margen de tolerancia para evitar cortar la cola), funciona de la manera esperada.

Respecto a la clasificación, se definieron dos clases trabajadas, bonito y cojinova. La clase “no pez” no se incluyó, pues antes de la clasificación el sistema realizó la detección. Los ajustes en el conjunto de imágenes final con el que el modelo entrenó. Se trabajó con una cantidad homogénea y balanceada para cada clase. Esto significa que se entrenó al modelo durante 100 épocas, con ciento seis (106) imágenes por cada clase, se realizó la validación paralela con veintiocho (28) imágenes y se hicieron las pruebas finales con veinte (20) imágenes adicionales por clase. Se obtuvieron los siguientes resultados:

- Durante el entrenamiento, el modelo convergió, aproximadamente, en la época n.º 30.
- Los valores de precisión y pérdida no se vieron afectados de forma negativa, es decir, no decayeron por más que el modelo había conseguido valores ideales. Estos valores son de  $\approx 0,1$  para la función de pérdida y 0,97 para el valor de precisión.
- En la fase de pruebas, se clasificó correctamente al 80 % de las imágenes pertenecientes al conjunto designado para pruebas.

En la segunda fase, se realizaron pruebas con especímenes de peces reales. Las clases escogidas fueron cojinova y bonito. Se trabajó con un espécimen por cada clase indicada. Los resultados obtenidos son los siguientes: respecto a la detección, se detectaron correctamente todos los peces presentados al sistema; respecto a la clasificación, el sistema clasificó correctamente el 80 % de las veces; sin embargo, en el 20 % de las pruebas realizadas con la clase cojinova se obtuvieron falsos positivos.

## 5.1 Discusión de resultados

Respecto a la primera fase de pruebas, se concluye que los resultados obtenidos durante la detección fueron satisfactorios porque alcanzaron el 90 % de exactitud. La solución planteada

para la detección de peces solo puede fallar si el pez tiene una forma distinta a la de un pez convencional o si el objeto a detectar no es un pez y tiene una forma similar a la de uno.

Respecto a los resultados de la tarea de clasificación, consideramos que son ideales. En términos de convergencia del modelo, se obtuvo convergencia en  $\approx$  al 80 % del proceso de entrenamiento. Esto se da porque el conjunto de datos es pequeño y, por la propia estructura y diseño del modelo, se consiguen valores ideales de manera rápida. Asimismo, la convergencia no se vio afectada por la continuidad de entrenamiento, en el resto de épocas, razón por la cual se muestra que el modelo es consistente.

Respecto a la precisión del modelo, plasmándola en términos sencillos, de cada diez imágenes capturadas de especímenes, ocho son clasificadas de manera correcta. Esto implica que los resultados obtenidos presentan un nivel elevado de precisión. De igual forma, realizando un análisis del porqué no es efectivo en su totalidad, concluimos que la inclusión de un número mayor de muestras representativas en el proceso de entrenamiento contribuiría a lograr el objetivo esperado.

Finalmente, de la segunda fase de pruebas se concluye que, respecto a la detección, se logró la detección satisfactoria de todos los peces reales en condiciones ideales de luminosidad. En lo que concierne a la tarea de clasificación, como se comentó, los falsos positivos que se obtienen son consecuencia de la similitud morfológica que tienen las clases estudiadas. El equipo de trabajo concluye que para optimizar este resultado es ideal contar con un conjunto de datos más extenso para el entrenamiento, de tal manera que el modelo de clasificación pueda generalizar mejor las clases de pescados incluidas.

## REFERENCIAS

- Aghdam, O. A. (s. f.). Deep Learning Features for Fish Classification, 3-5.
- Brown, M., y Lowe, D. G. (2002). Invariant Features from Interest Point Groups. *British Machine Vision Conference*. doi:10.1.1.1.8475
- Calvo Dopico, D. (2015). Implantación de la trazabilidad y su relación con la calidad: marco conceptual y retos estratégicos. Aplicación al sector pesquero. *Economía Agraria y Recursos Naturales*, 15(1), 79-98. doi:10.7201/earn.2015.01.05.a
- Caribbean Regional Fisheries Mechanism (CRFM). (2016). Manual de sistemas de trazabilidad del pescado y productos pesqueros. *Instituto Interamericano de Cooperación para la Agricultura, CRFM Publicación Especial*, (13).
- Chen, G., Sun, P., y Shang, Y. (2017). Automatic fish classification system using deep learning. *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*, 24-29. doi:10.1109/ICTAI.2017.00016

- Chincha, R., y Tian, Y. (2011, November). Finding objects for blind people based on SURF features. *2011 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW)*. (526-527). IEEE.
- Edelman, S., Intrator, N., y Poggio, T. (1997). Complex cells and object recognition. *Computer Science*.
- FAO. (2018). El estado mundial de la pesca y la acuicultura 2018. Cumplir los objetivos de desarrollo sostenible. Roma. Licencia: CC BY-NC-SA 3.0 IGO
- Fishing for a Future - Nature Works Everywhere. (s. f.). Recuperado de <https://www.natureworkseverywhere.org/resources/fishing-for-a-future/>
- Fouad, M. M., Zawbaa, H. M., El-Bendary, N., y Hassanien, A. E. (2013). Automatic Nile Tilapia fish classification approach using machine learning techniques. *13th International Conference on Hybrid Intelligent Systems (HIS 2013)*, 173-178. doi:10.1109/HIS.2013.6920477
- Frost, J., Geisler, T., y Mahajan, A. (s. f.). Monitoring illegal fishing through image classification.
- Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36. Recuperado de <https://www.rctn.org/bruno/public/papers/Fukushima1980.pdf>
- Harris, C., y Stephens, M. (1988). A combined corner and edge detector. *Proceedings of the Alvey Vision Conference 1988*. doi:10.5244/C.2.23
- He, K., Gkioxari, G., Dollár, P., y Girshick, R. (2017). Mask R-CNN. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2980-2988. doi:10.1109/ICCV.2017.322
- He, K., Zhang, X., Ren, S., y Sun, J. (2015). Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778. doi:10.1109/CVPR.2016.90
- Huang, G., Liu, Z., van der Maaten, L., y Weinberger, K. Q., (2017). Densely connected convolutional networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2261-2269. doi:10.1109/CVPR.2017.243
- Josifovski, J. (s. f.). Object recognition sift vs convolutional neural networks. Recuperado de [https://tams.informatik.uni-hamburg.de/lehre/2015ws/seminar/ir/pdf/slides/JosipJosifovskiObject\\_Recognition\\_SIFT\\_vs\\_Convolutional\\_Neural\\_Networks.pdf](https://tams.informatik.uni-hamburg.de/lehre/2015ws/seminar/ir/pdf/slides/JosipJosifovskiObject_Recognition_SIFT_vs_Convolutional_Neural_Networks.pdf)
- Juan, L., y Gwun, O. (2009). A comparison of SIFT, PCA-SIFT and SURF. *International Journal of Image Processing (IJIP)*, 3(4), 143-152.
- Karami, E., Prasad, S., y Shehata, M. (2015). Image matching using SIFT, SURF, BRIEF and ORB: Performance comparison for distorted images. *Newfoundland Electrical and Computer Engineering Conference*.

- Krizhevsky, A., Sutskever, I., y Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances In Neural Information Processing Systems*, 1-9. doi/10.5555/2999134.2999257
- Kruger, N., Janssen, P., Kalkan, S., Lappe, M., Leonardis, A., Piater, J., ... Wiskott, L. (2013). Deep hierarchies in the primate visual cortex: What can we learn for computer vision? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8), 1847-1871. doi:10.1109/TPAMI.2012.272
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., y Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4), 541-551. doi:10.1162/neco.1989.1.4.541
- Lowe, D. G. (1999). Object Recognition from Local Scale-Invariant Features (*SIFT*). *International Conference on Computer Vision*. doi:10.1109/ICCV.1999.790410
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91-110. doi:10.1023/B:VISI.0000029664.99615.94
- Mahajan, A., y Perkins, A. Whale Detection and Identification from Aerial Photography.
- Matai, J., Kastner, R., Cutter, G. R., y Demer, D. A. (2012). Automated techniques for detection and recognition of fishes using computer vision algorithms. *NOAA Technical Memorandum*, 35-37.
- Meng, Y., y Tiddeman, B. (2006). Implementing the Scale Invariant Feature Transform (SIFT) Method. Citeseer.
- Nadarajan, G., Chen-Burger, J., Fisher, R., y Spampinato, C. (2011). A flexible system for automated composition of intelligent video analysis. *ISPA 2011-7th International Symposium on Image and Signal Processing and Analysis*, 259-264
- Piccardi, M. (2004). Background subtraction techniques: a review. *2004 IEEE International Conference on Systems, Man and Cybernetics*. doi:10.1109/ICSMC.2004.1400815
- Ren, S., He, K., Girshick, R., y Sun, J. (2016). Faster R-CNN: Towards real-time object detection with region proposal networks. 1-14. doi:10.1109/TPAMI.2016.2577031
- Shujuan, S., Zhize, X., Xingang, W., Guan, H., Wenqi, W., y De, X. (2015, May). Real-time vehicle detection using Haar-SURF mixed features and gentle AdaBoost classifier. *The 27th Chinese Control and Decision Conference (2015 CCDC)*, 1888-1894. IEEE.
- Simonyan, K., y Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

- Sobral, A., y Vacavant, A. (2014). A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding*, 122, 4-21. doi:10.1016/j.cviu.2013.12.005
- Sociedad Nacional de Pesquería (2018). Exportaciones pesqueras. Recuperado de <https://www.snp.org.pe/wp-content/uploads/2019/02/REPORTE-DE-LAS-EXPORT-PESQUERAS-A%C3%91O-2018.pdf>
- Suleiman, A., Chen, Y. H., Emer, J., y Sze, V. (2017). Towards closing the energy gap between HOG and CNN features for embedded vision (invited paper). *Proceedings - IEEE International Symposium on Circuits and Systems*. doi:10.1109/ISCAS.2017.8050341
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., y Rabinovich, A. (2014). Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1-9. doi:10.1109/CVPR.2015.7298594
- Tareen, S. A., y Saleem, Z. (2018). A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK. *2018 International Conference on Computing, Mathematics and Engineering Technologies (ICOMET)*, 1-10. doi:10.1109/ICOMET.2018.8346440
- Treiber, M. A. (2010). An introduction to object recognition: selected algorithms for a wide variety of applications. *Springer Science & Business Media*.
- Vedaldi, A. (2006). An implementation of SIFT detector and descriptor. University of California at Los Angeles, 7.
- Zeiler, M. D., y Fergus, R. (2014). LNCS 8689 - Visualizing and understanding convolutional networks. *Computer Vision, ECCV 2014 - 13th European Conference, Proceedings*. Recuperado de <https://cs.nyu.edu/~fergus/papers/zeilerECCV2014.pdf>