**Escuela Técnica Superior de Ingeniería Universidad de Huelva**

Grado de Ingeniería Informática

**Trabajo Fin de Grado**

Aplicaciones de estrategias de Deep Learning para la detección de animales en imágenes de fototrampeo

Imagen que contiene Texto

Descripción generada automáticamenteUn dibujo con letras

El contenido generado por IA puede ser incorrecto.Forma

El contenido generado por IA puede ser incorrecto.

Carlos García Silva

Mayo 2025

TRABAJO DE FIN DE GRADO – ESCUELA SUPERIOR DE INGENIERÍA

**Aplicaciones de estrategias de Deep Learning para la detección de animales en imágenes de fototrampeo**

Carlos García Silva Huelva, Mayo 2025

Resumen

La importancia de un futuro sostenible se refleja cada día con mayor claridad en nuestra sociedad. Es por ello por lo que los estudios de biodiversidad han cobrado un papel fundamental en la compresión, conservación y valoración de la vida en la Tierra.

Una de las técnicas ampliamente utilizada para la realización de dichos estudios es la técnica del foto-trampeo. Se basa en el despliegue de cámaras automáticas en el entorno a estudiar para poder capturar imágenes de animales en su hábitat natural. Uno de sus principales problemas es la gran facilidad para generar grandes volúmenes de imágenes, en las que realizar un análisis manual puede resultar costoso y laborioso.

Este Trabajo de Fin de Grado presenta una red neuronal convolucional capaz de clasificar imágenes de foto-trampeo para identificar aquellas donde existe presencia de animales. El sistema propuesto ha sido desarrollado mediante el uso de la técnica de transfer learning, empleando un modelo previamente entrenado con los pesos de ImageNet, basado en la arquitectura EfficientNet-B5.

El experimento se realizó utilizando una colección de 31.670 imágenes obtenidas de cámaras de foto-trampeo, de las cuales 21.670 presentaban evidencia de presencia animal. El proceso de entrenamiento, validación y evaluación de la red se llevó a cabo utilizando una proporción del 70%, 15% y 15% del conjunto de imágenes, respectivamente.

*Párrafo de resultados* [TO-DO]

*Párrafo final con conclusiones y valoración* [TO-DO]

**Palabras clave:** [TO-DO]

BACHELOR’S THESIS – HIGHER TECHNICAL SCHOOL OF ENGINEERING

**Applications of Deep Learning strategies for the detection of animals in cameratraps images**

Carlos García Silva Huelva, May 2025

Abstract

The importance of a sustainable future is reflected more and more clearly in our society. This is why biodiversity studies have taken on a fundamental role in the understanding, conservation and valuation of life on Earth.

One of the techniques widely used to carry out such studies is the photo-trapping technique. It is based on the deployment of automatic cameras in the environment to be studied in order to capture images of animals in their natural habitat. One of its main problems is that it is very easy to generate large volumes of images, where manual analysis can be costly and time-consuming.

This Final Degree Project presents a convolutional neural network capable of classifying photo-trapping images to identify those where animals are present. The proposed system has been developed using the transfer learning technique, employing a model previously trained with ImageNet weights, based on the EfficientNet-B5 architecture.

The experiment was conducted using a collection of 31.670 images obtained from photo-trapping cameras, of which 21.670 showed evidence of animal presence. The network training, validation and evaluation process was carried out using a proportion of 70%, 15% and 15% of the image set, respectively.

*Párrafo de resultados* [TO-DO]

*Párrafo final con conclusiones y valoración* [TO-DO]

**Keywords:** [TO-DO]

Índice general

[Resumen II](#_Toc195169191)

[Abstract IV](#_Toc195169192)

[Índice de figuras VII](#_Toc195169193)

[1 Propuesta de Proyecto 1](#_Toc195169194)

[1.1 Motivación 1](#_Toc195169195)

[1.2 Objetivos 2](#_Toc195169196)

[1.3 Competencias 3](#_Toc195169197)

[1.4 Hardware y Sofware 3](#_Toc195169198)

[1.5 Organización de la memoria 4](#_Toc195169199)

[2 Introducción y estado del arte 6](#_Toc195169200)

[3 Materiales 7](#_Toc195169201)

[3.1 Base de datos 7](#_Toc195169202)

[3.2 Conjunto de datos 8](#_Toc195169203)

[3.3 Métricas de evaluación 8](#_Toc195169204)

[4 Metodología 9](#_Toc195169205)

[4.1 Arquitectura de la red 9](#_Toc195169206)

[4.2 Hiperparámetros 9](#_Toc195169207)

[4.3 Fase de entrenamiento 9](#_Toc195169208)

[5 Resultados y discusión 10](#_Toc195169209)

[5.1 Resultados del conjunto de prueba 10](#_Toc195169210)

[5.2 Análisis y discusión 10](#_Toc195169211)

[6 Conclusiones y trabajos futuros 11](#_Toc195169212)

[6.1 Conclusiones técnicas 11](#_Toc195169213)

[6.2 Trabajos futuros 11](#_Toc195169214)

[6.3 Valoración personal 11](#_Toc195169215)

Índice de figuras

[Figura 1: Distribución de las clases en el dataset original 7](#_Toc195169020)

[Figura 2: Distribución binaria del dataset original 8](#_Toc195169021)

**No se encuentran elementos de tabla de ilustraciones.**

# Propuesta de Proyecto

En el presente capítulo introductorio, se exponen las motivaciones que han fundamentado la realización de este trabajo, los objetivos que se han establecido y el sistema implementado para su consecución. Asimismo, se detalla el desarrollo de dichos objetivos y las diversas tecnologías empleadas.

## Motivación

En años recientes, se han observado significativos avances en el ámbito de la tecnología, lo cual ha generado un impacto sustancial en la sociedad contemporánea. En cierta medida, se puede afirmar que la dependencia hacia esta área es una realidad en el contexto cotidiano. Diversos analistas consideran que la actual situación representa una nueva revolución industrial. En este sentido, se puede afirmar que, del mismo modo que sucedió con la industria textil en la primera revolución industrial, la electricidad en la segunda y la electrónica en la tercera, hoy en día se puede constatar que la informática se erige como uno de los pilares que han permitido identificar a esta nueva revolución como la denominada Industria 4.0.

En el campo de la informática, se ha observado un aumento significativo en la implementación de algoritmos y técnicas de inteligencia artificial en los últimos años. Este fenómeno se atribuye al incremento en la capacidad de procesamiento y al vasto volumen de datos disponibles en la actualidad, en contraste con la limitada disponibilidad de hace unos años. En particular, nos referimos a los algoritmos de aprendizaje profundo (Deep Learning)[[1]](#footnote-1), que se fundamentan principalmente en redes neuronales artificiales.

Dentro del ámbito de la inteligencia artificial, los algoritmos han experimentado una notable evolución en su capacidad de procesamiento de datos. La implementación de estos algoritmos se lleva a cabo durante el proceso de entrenamiento, mediante la utilización de vastas cantidades de datos. Además, estos algoritmos se ejecutan en sistemas con capacidades de cómputo que, hasta hace una década, se consideraban inalcanzables para las máquinas. Los resultados obtenidos recientemente han superado las expectativas, logrando niveles de rendimiento que anteriormente eran impensables para las máquinas y, en algunos casos, incluso superando a los humanos en ciertas tareas. Este avance representa una significativa apertura a la capacidad de abordar nuevos y complejos problemas que, anteriormente, solo podían resolverse mediante el ingenio humano.

El propósito de este análisis es desarrollar e implementar una red neuronal convolucional (CNN)[[2]](#footnote-2) con la capacidad de clasificar imágenes obtenidas mediante fototrampeo, diferenciando entre aquellas que contienen presencia de animales y las que no. Este sistema de clasificación se aplicará sobre conjuntos de datos reales con el fin de explorar su precisión y eficacia en el contexto del monitoreo de la fauna silvestre.

Para evaluar el desempeño del modelo propuesto, se compararán los resultados obtenidos con los de MegaDetector[[3]](#footnote-3), un modelo de detección de fauna ampliamente utilizada y desarrollada por Microsoft. A pesar de que MegaDetector está orientado a la detección, sus salidas pueden reinterpretarse como una forma de clasificación binaria (presencia animal/vacía), lo que lo convierte en una referencia adecuada para establecer una comparación de rendimiento. Esta comparación permitirá analizar las ventajas y limitaciones dentro de un marco de conservación y automatización del análisis de imágenes de fototrampeo.

Este proyecto constituye una oportunidad para profundizar en el conocimiento y la aplicación de técnicas avanzadas de Deep Learning, que no se abordan en el plan de estudios del Grado en Ingeniería Informática. Además, posee una clara relevancia práctica en el ámbito de la conservación de la biodiversidad. La implementación de CNN en la clasificación de imágenes de fototrampeo permite optimizar el monitoreo de la fauna, facilitando el análisis de datos con mayor eficiencia y precisión. Este avance no solo contribuye al desarrollo de nuevas metodologías en el procesamiento de imágenes, sino que también abre la posibilidad de generar herramientas de apoyo en estudios ecológicos y en la preservación de especies, lo que destaca la importancia de la inteligencia artificial en la resolución de problemas del mundo real.

## Objetivos

En el marco de la presente investigación, se han planteado dos objetivos primordiales a alcanzar:

* **Adquisición de conocimiento:** Introducción y estudio teórico de las técnicas de Deep Learning, con el propósito de comprender su naturaleza, su evolución hasta la actualidad y los fundamentos necesarios para su aplicación en el problema específico de interés.
* **Implementación:** En segundo lugar, se implementará un sistema basado en CNN que permitirá llevar a cabo una clasificación eficaz de la presencia o no, de animales en imágenes de fototrampeo.

Para una mayor profundización en el tema, se propone una subdivisión de los objetivos principales en los siguientes puntos: En primer lugar, es necesario realizar una exhaustiva revisión bibliográfica y un análisis de los conceptos teóricos básicos acerca del Deep Learning. En segundo lugar, se obtendrá, analizará y preparará un conjunto de imágenes[[4]](#footnote-4) para el entrenamiento, la validación y la evaluación de los modelos implementados. En tercer lugar, se procederá al diseño, implementación y evaluación de varios modelos entrenados para la detección de animales en imágenes de foto-trampeo. Por último, se llevará a cabo una búsqueda e implementación de métricas para la evaluación de los modelos, que permitirá cuantificar objetivamente la calidad de los resultados obtenidos.

## Competencias

El desarrollo de este proyecto ha permitido la aplicación y el refuerzo de diversas competencias del Grado en Ingeniería Informática, particularmente aquellas vinculadas con el aprendizaje computacional y el diseño e implementación de sistemas basados en inteligencia artificial. En particular, se ha focalizado en el estudio y la aplicación de las CNN para la clasificación de imágenes de fototrampeo. Este proceso ha implicado una revisión exhaustiva del estado del arte en Deep Learning, así como la experimentación con modelos diseñados específicamente para la detección de fauna en imágenes digitales.

En contraste, la imperativa de operar con datasets de un entorno real ha posibilitado la evolución de competencias en la extracción automática de información y filtrado de imágenes para potenciar la precisión de los modelos. En este sentido, el trabajo no solo ha servido como un ejercicio de profundización en técnicas de Deep Learning que no se abordan con detalle en el plan de estudios, sino también ha permitido aplicar estos conocimientos a un caso práctico de relevancia en el ámbito de la conservación de la fauna, evidenciando la importancia del aprendizaje computacional en la solución de problemas en situaciones reales.

En resumen, las principales competencias a desarrollar son la capacidad para conocer y desarrollar técnicas de aprendizaje computacional y diseñar e implementar aplicaciones y sistemas que las utilicen, incluyendo las dedicadas a extracción automática de información y conocimiento a partir de grandes volúmenes de datos.

## Hardware y Sofware

Para poder alcanzar las metas anteriormente establecidas de manera efectiva, resulta necesario el uso de un sistema cuyo hardware posea la capacidad de ejecutar algoritmos de Deep Learning. Además, dicho sistema debe estar equipado con un software especializado en el diseño y creación de modelos que hagan uso de estos algoritmos.

En lo que respecta al software empleado, la implementación del modelo se realizará en el lenguaje de programación Python 3.8 debido a su amplia popularidad en aplicaciones de aprendizaje automático. EL framework seleccionado para la ejecución de este proyecto es TensorFlow[[5]](#footnote-5), debido a su exhaustiva documentación y activa comunidad de usuarios, lo que facilitará la implementación del proyecto. Desde su versión 2.0, TensorFlow incorpora de forma nativa la librería Keras[[6]](#footnote-6), que permite la creación de redes neuronales desde un alto nivel gracias a su estructura de capas de abstracción. Esta integración proporciona una metodología eficiente para la concepción de modelos de Deep Learning, permitiendo descender en la abstracción solo cuando sea necesario y de forma más precisa.

Para la implementación y el entrenamiento del modelo de clasificación se utilizarán aplicaciones como Jupyter Notebook[[7]](#footnote-7) y Visual Studio Code[[8]](#footnote-8), junto con el entorno de desarrollo proporcionado por Anaconda[[9]](#footnote-9), que nos permitirá instalar y gestionar fácilmente las librerías necesarias, como Tensorflow, OpenCV[[10]](#footnote-10), Numpy[[11]](#footnote-11) y otras más.

Para la implementación del proyecto, se ha utilizado un MacBook Pro de 14 pulgadas, equipado con el chip Apple M3 (ver Figura 2), 16 GB de memoria unificada y un almacenamiento SSD de 512 GB. Este dispositivo ha permitido la ejecución eficiente de los experimentos, gracias a su CPU de 8 núcleos, optimizada para tareas de alto rendimiento y eficiencia, así como su GPU de 10 núcleos, que incorpora tecnologías avanzadas como Dynamic Caching, Mesh Shading y Ray Tracing acelerado por hardware. Además, la librería TensorFlow ha experimentado actualizaciones que han posibilitado su compatibilidad nativa con la nueva arquitectura de procesadores Apple Silicon, lo que ha conducido a un notable incremento en su rendimiento.

## Organización de la memoria

Además del primer capítulo, en el que se presenta la propuesta, la motivación y los objetivos del trabajo, así como las competencias desarrolladas y los recursos hardware y software empleados, esta memoria se organiza en los siguientes capítulos:

* **Capítulo 2 Introducción y estado de arte:** Se presenta un análisis exhaustivo de la implementación de técnicas de Deep Learning en el ámbito de la clasificación de imágenes. Este capítulo examina estudios previos y tecnologías relevantes en el contexto del reconocimiento de fauna mediante foto-trampeo.
* **Capítulo 3 Materiales:** Se aborda la descripción de los materiales empleados en el desarrollo del modelo. En este capítulo se describen los datos utilizados en la implementación del modelo, incluyendo la base de datos empleada, los conjuntos de datos generados para el entrenamiento, validación y evaluación; y las métricas utilizadas para medir el desempeño del sistema.
* **Capítulo 4 Metodología:** Como su propio nombre indica, se centra en la metodología empleada, detallando las estrategias implementadas en el desarrollo del modelo. En este sentido, se aborda la selección de la arquitectura de la red neuronal, los hiperparámetros utilizados y el proceso de entrenamiento del modelo.
* **Capítulo 5 Resultados y discusión:** Presenta los resultados obtenidos en el conjunto de evaluación, analizando su rendimiento en términos de las métricas establecidas y discutiendo las fortalezas y limitaciones del enfoque adoptado.
* **Capítulo 6 Conclusiones y trabajos futuros:** Se presentan las conclusiones técnicas extraídas del estudio, se identifican posibles mejoras y líneas de trabajo futuro, y se incluye una valoración personal sobre la experiencia adquirida durante la realización del proyecto.

La memoria prosigue con los capítulos destinados a la bibliografía y referencias, en el cual, como su propio nombre indica se efectúan las referencias de los artículos, libros y recursos empleados para la elaboración del trabajo.

Además de los capítulos principales, la memoria contiene una serie de anexos teóricos adicionales que profundizan en los conceptos fundamentales del trabajo. Se encuentra organizado de la siguiente manera:

* **Anexo teórico A: Introducción al Deep Learning:** Se presentan los principios básicos del Deep Learning, destacando su evolución y relevancia en la actualidad.
* **Anexo teórico B: Redes Neuronales:** En este apartado se abordan los fundamentos del funcionamiento de las redes neuronales, desde su desarrollo hasta las diferentes etapas del entrenamiento e inferencia, detallando aspectos como la configuración de hiperparámetros y el control del proceso de entrenamiento.
* **Anexo teórico C: Redes Neuronales Convolucionales:** Se analiza la estructura y funcionamiento de las CNN, describiendo los tipos de capas utilizadas, las arquitecturas más populares y la arquitectura específica empleada en este trabajo.
* **Anexo teórico D: MegaDetector:** Finalmente se presenta MegaDetector, utilizado para el preprocesamiento de las imágenes de foto-trampeo utilizadas como entrada para nuestro modelo clasificador, explicando su funcionamiento y su integración dentro del sistema desarrollado.

# Introducción y estado del arte

[TO-DO]

# Materiales

En el presente capítulo se aborda la descripción de los recursos empleados durante el desarrollo del trabajo, iniciando con la base de datos que sirve como punto de acceso al modelo, detallando su procedencia, clases disponibles y distribución de instancias. Posteriormente, se especifica la división de los datos en conjuntos de entrenamiento, validación y prueba, así como los criterios seguidos para dicha partición. Por último, se enumeran las métricas de evaluación empleadas para valorar el rendimiento del modelo propuesto, tanto durante la fase de entrenamiento como en los experimentos finales.

## Base de datos

Para la implementación de este modelo, capaz de clasificar imágenes con o sin presencia animal, se ha proporcionado un conjunto de datos compuesto por imágenes de fototrampeo previamente clasificadas.

Sin embargo, esta categorización inicial no se alineaba con nuestro objetivo, ya que se basaba en la clasificación taxonómica de las especies presentes en las imágenes. Por consiguiente, fue necesario realizar una adaptación minuciosa para ajustarla a las características específicas de nuestro caso de estudio.

Originalmente, se nos proporcionó un conjunto de datos que contenía 19 clases taxonómicas distintas, sumando un total de 31.670 imágenes, como se muestra en laFigura 1. Sin embargo, dado que nuestro problema de clasificación se basa en una dicotomía binaria, mantendremos la clase denominada *Vacía* para indicar la ausencia de animales en la imagen.

A pie chart with numbers and text

AI-generated content may be incorrect.Figura 1: Distribución de las clases en el dataset original

Gráfico

El contenido generado por IA puede ser incorrecto.Por otro lado, fusionaremos las demás clases, correspondientes a diferentes especies, para formar una única categoría llamada Animal. Como se evidencia en la Figura 2, se obtiene la proporción de aproximadamente 30% del conjunto de imágenes con ausencia de animales y un % con presencia. Estos datos ya nos adelantan que tendremos que trabajar con técnicas que tengan en cuenta un desbalance entre las clases.

Figura 2: Distribución binaria del dataset original

Mediante el análisis de la composición de las imágenes, se evidencia la presencia de una diversidad de elementos. En primer lugar, se observa que algunas de las imágenes fueron capturadas durante el día (como laFigura 4), en las cuales el animal ocupa una porción significativa del plano. En segundo lugar, se identifica que, en aquellas imágenes capturadas durante el día, puede ser difícil discernir la ubicación del animal, ya sea debido a su camuflaje en el entorno (como se muestra en laFigura 3) o a la comparación de su tamaño con el encuadre del plano (como se observa en laFigura 5).



Figura 3: Ejemplo de imagen diurna con animal camuflado resaltado



Figura 4: Ejemplo de imagen diurna con presencia animal

En oposición a las imágenes previamente mencionadas, se presentan aquellas que fueron capturadas durante el período nocturno, lo que implica que fueron tomadas en condiciones de baja luminosidad y, por consiguiente, con una marcada ausencia de color en la fotografía. En este tipo de imágenes se observan situaciones similares al caso anterior, donde el animal puede cubrir una porción significativa del encuadre (ver Figura 6) o también animales camuflados en el entorno nocturno (ver Figura 7).



Figura 5: Ejemplo de imagen nocturna con animal camuflado resaltado



Figura 6: Ejemplo de imagen nocturna con presencia animal

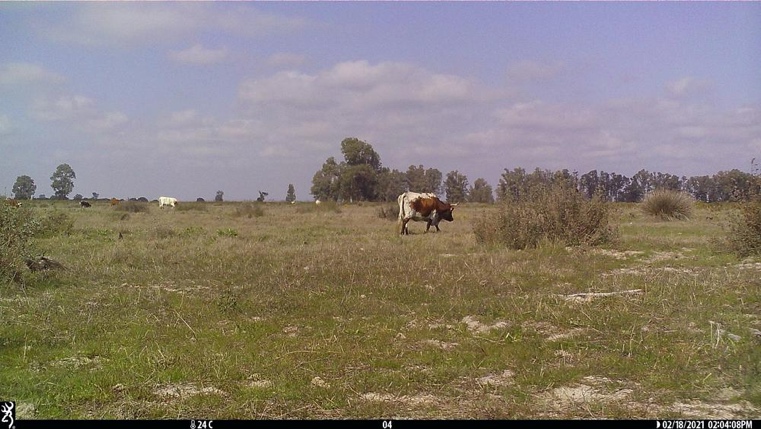


Figura 7: Ejemplo de imagen diurna con presencia animal en un encuadre muy abierto

La disponibilidad de un conjunto de datos de fototrampeo que abarque una amplia gama de situaciones, composiciones, condiciones ambientales y tipos de terreno constituye una ventaja significativa en el entrenamiento de modelos de aprendizaje profundo. Además, la presencia de diferentes niveles de iluminación, ángulos de captura, fondos y especies animales permitirá al modelo aprender patrones más robustos y desarrollar una mayor capacidad para enfrentarse a casos complejos.

No obstante, durante el análisis de las diferentes casuísticas que acabamos de exponer, se observó que las imágenes pertenecientes a la clase original denominada como “dudosa”, suponían unas situaciones extremas y, por tanto, demasiado complejas para que un ser humano pudiera catalogarlas, de hecho, el propio nombre de la clase lo autodescribe. Dichos hallazgos apuntan hacia la posibilidad de que surgieran problemas relacionados con su uso, por lo que su implementación ha sido descartada.

Gráfico

El contenido generado por IA puede ser incorrecto.Además de la eliminación de la clase dudosa, se observó la presencia de rutas de imágenes duplicadas, por lo que se llevó a cabo una meticulosa limpieza, teniendo como resultado una colección con un total de 28.567 imágenes, distribuidas según se muestra en la Figura 8.

Figura 8: Distribución binaria del dataset final

## Conjunto de datos

Una vez concluido el análisis de los datos obtenidos en el paso anterior, se procede a la división del conjunto de datos en tres subconjuntos. Dividir el conjunto de datos en tres subconjuntos es una práctica fundamental en el desarrollo de modelos de redes neuronales convolucionales. Si bien se recomienda como mínimo la división de dos subconjuntos, hacerlo en tres permite una mejor evaluación exhaustiva y objetiva del rendimiento del modelo, contribuyendo así a su optimización y mejora continua.

Como se explica con mayor detalle en la **[TOD-DO]** sección 2.2.1 Etapa de entrenamiento del anexo teórico **[TO-DO]**, el subconjunto de entrenamiento se emplea para ajustar los pesos de la red durante el proceso de aprendizaje. Por su parte, el de validación supervisa el comportamiento del modelo durante el entrenamiento y permite tomar decisiones como el uso de early stopping[[12]](#footnote-12), la elección de hiperparámetros o la detección de posibles problemas de sobreajuste. Por último, el conjunto de prueba proporciona una evaluación final, imparcial e independiente del modelo ya entrenado, midiendo su capacidad real de generalización sobre datos nunca vistos.

La implementación de este procedimiento garantiza la integridad de los resultados obtenidos, al evitar la influencia de los datos empleados durante el proceso de entrenamiento. De este modo, se consigue una representación más precisa del rendimiento esperado en escenarios del mundo real.

En nuestro caso, se implementó una estrategia de subdivisión que asignó el 70 % de las imágenes al subconjunto de entrenamiento, el 15 % al de validación y el 15 % restante al de prueba. En el proceso de división, se preservó la proporción de imágenes con y sin presencia animal en los conjuntos de validación y test, manteniendo la misma distribución observada en el conjunto de datos completo.

Tabla 1: Número de imágenes de cada clase en cada división del dataset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Conjunto | Porcentaje de imágenes disponibles | Número total de imágenes | Número de imágenes con presencia animal | Número de imágenes vacías |
| Entrenamiento | 70% | 19.995 | 13.230 | 6.765 |
| Validación | 15% | 4.286 | 2.836 | 1.450 |
| Prueba | 15% | 4.286 | 2.836 | 1.450 |

Se evidencia un desbalanceo entre las clases, con una predominancia de casos positivos que alcanzan aproximadamente el 66%, en contraste con los casos negativos que representan el 34% del total.

## Métricas de evaluación

Con el propósito de evaluar la eficacia del modelo de clasificación binaria propuesto, se ha implementado un conjunto de métricas que facilitan la observación de su comportamiento desde múltiples perspectivas.

* **Accuracy:** mide el porcentaje global de predicciones correctas.
* **Loss:** refleja el error durante el entrenamiento y la validación
* **Precision:** ofrece información más detallada sobre la calidad de las predicciones positivas, indica la proporción de predicciones positivas que resultaron ser correctas.
* **Recall:** también ofrece información detallada sobre la calidad de las predicciones positivas, en este caso, la proporción de instancias positivas reales que fueron identificadas correctamente.
* **Specificity:** también ofrece información detallada sobre la calidad de las predicciones negativas, en este caso, la proporción de instancias negativas reales que fueron identificadas correctamente.
* **F1-Score:**  integra ambas métricas (precision y recall) en una única medida equilibrada.
* **Matriz de confusión:** para visualizar de forma detallada la distribución de aciertos y errores en las predicciones.
* **AUC:** Área Bajo la Curva ROC evalúa la capacidad del modelo para distinguir entre clases.
* **Curva ROC:** complementada con la métrica anterior, en este caso ilustra gráficamente la capacidad de distinguir entre clases.

Todas estas métricas se encuentran mejor definidas en el **[TO-DO]** apartado 2.2.1 Etapa de entrenamiento del anexo teórico **[TO-DO]**.

# Metodología

[TO-DO]

## Arquitectura de la red

[TO-DO]

## Hiperparámetros

[TO-DO]

## Fase de entrenamiento

[TO-DO]

# Resultados y discusión

[TO-DO]

## Resultados del conjunto de prueba

[TO-DO]

## Análisis y discusión

[TO-DO]

# Conclusiones y trabajos futuros

[TO-DO]

## Conclusiones técnicas

[TO-DO]

## Trabajos futuros

[TO-DO]

## Valoración personal

[TO-DO]

1. Debido a la popularidad del término Deep Learning durante el resto del trabajo lo utilizaremos en sustitución del término aprendizaje profundo. [↑](#footnote-ref-1)
2. Siglas del término en inglés Convolutional Neural Network. Por motivos de simplicidad, durante el resto del trabajo utilizaremos dichas siglas en sustitución del término red neuronal convolucional. [↑](#footnote-ref-2)
3. Enlace al repositorio oficial del modelo de detección:

   <https://github.com/microsoft/CameraTraps/tree/main> [↑](#footnote-ref-3)
4. Debido a su popularidad, a partir de este momento, se utilizará el término en inglés *dataset* en sustitución del término conjunto de datos/imágenes. [↑](#footnote-ref-4)
5. [https://www.tensorflow.org](https://www.tensorflow.org/?hl=es-419) [↑](#footnote-ref-5)
6. <https://keras.io/> [↑](#footnote-ref-6)
7. <https://jupyter.org/> [↑](#footnote-ref-7)
8. <https://code.visualstudio.com/> [↑](#footnote-ref-8)
9. <https://www.anaconda.com/> [↑](#footnote-ref-9)
10. <https://opencv.org/> [↑](#footnote-ref-10)
11. <https://numpy.org/> [↑](#footnote-ref-11)
12. Técnica de regularización utilizada en entrenamientos de modelos de Machine Learning, monitoreando una métrica seleccionada para la ejecución del entreno cuando ésta deja de mejorar. [↑](#footnote-ref-12)