

Clasificación de especies de mariposas y polillas mediante aprendizaje profundo



Universitat
Oberta
de Catalunya

María del Mar Ruiz Alcántara

MU Ingeniería Informática
Área de Inteligencia Artificial

Nombre Tutor/a de TF

Joan M. Nuñez do Rio

**Profesor/a responsable de la
asignatura**

Carles Ventura Royo

14/01/2024



Esta obra está sujeta a una licencia de
Reconocimiento-NoComercial-
SinObraDerivada [3.0 España de Creative
Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Clasificación de especies de mariposas y polillas mediante aprendizaje profundo</i>
Nombre del autor:	<i>María del Mar Ruiz Alcántara</i>
Nombre del consultor/a:	<i>Juan M. Nuñez do Rio</i>
Nombre del PRA:	<i>Carles Ventura Royo</i>
Fecha de entrega (mm/aaaa):	<i>01/2024</i>
Titulación o programa:	MU Ingeniería Informática
Área del Trabajo Final:	<i>Inteligencia Artificial</i>
Idioma del trabajo:	<i>Castellano</i>
Palabras clave	<i>deep learning, CNN, image classification</i>
Resumen del Trabajo	
<p>La automatización de la clasificación de imágenes tiene aplicaciones extensas y significativas en diversos campos. Este estudio se enfoca específicamente en la clasificación de 100 especies distintas de mariposas y polillas, una tarea esencial para entender mejor la biodiversidad y contribuir a esfuerzos de conservación. Utilizando técnicas avanzadas de aprendizaje profundo, el proyecto busca demostrar cómo la inteligencia artificial puede agilizar y mejorar la precisión en la identificación de estas especies.</p> <p>El conjunto de datos empleado en este estudio incluye imágenes de mariposas y polillas, cada una etiquetada con su respectiva especie. Se configuraron y ajustaron varios modelos de redes neuronales convolucionales (CNN) para esta tarea. Tras una comparación exhaustiva de los resultados, se concluyó que los modelos preentrenados superan significativamente al modelo secuencial en términos de precisión y eficiencia. Este hallazgo destaca la importancia del aprendizaje transferido y la adaptabilidad de los modelos preentrenados en el campo de la clasificación de imágenes.</p> <p>Como conclusión del estudio, se evidencia que un modelo de red neuronal bien entrenado puede ser extremadamente eficaz en la clasificación de especies de mariposas y polillas, optimizando así la labor humana y reduciendo el consumo de recursos. Este proyecto no solo confirma que las redes neuronales son capaces de abordar satisfactoriamente este desafío, sino que también abre la puerta a futuras aplicaciones prácticas en campos como la ecología, la conservación de la biodiversidad y la investigación científica.</p>	

Abstract

The automation of image classification has extensive and significant applications in various fields. This study specifically focuses on the classification of 100 different species of butterflies and moths, a crucial task for better understanding biodiversity and contributing to conservation efforts. Using advanced deep learning techniques, the project aims to demonstrate how artificial intelligence can streamline and improve the accuracy of identifying these species.

The dataset used in this study includes images of butterflies and moths, each labeled with its respective species. Various convolutional neural network (CNN) models were configured and adjusted for this task. After a comprehensive comparison of the results, it was concluded that pretrained models significantly outperform the sequential model in terms of precision and efficiency. This finding highlights the importance of transfer learning and the adaptability of pretrained models in the field of image classification.

As conclusion of the study, it is evident that a well-trained neural network model can be extremely effective in classifying species of butterflies and moths, thus optimizing human labor and reducing resource consumption. This project not only confirms that neural networks are capable of satisfactorily addressing this challenge, but it also opens the door to future practical applications in fields such as ecology, biodiversity conservation, and scientific research.

Índice

1.	Introducción.....	1
1.1.	Contexto y justificación del Trabajo.....	1
1.2.	Objetivos del Trabajo	2
1.3.	Impacto en sostenibilidad, ético-social y de diversidad.....	4
1.4.	Enfoque y método seguido	5
1.5.	Planificación del Trabajo	6
1.6.	Breve resumen de productos obtenidos.....	9
1.7.	Breve descripción de los otros capítulos de la memoria	9
2.	Estado del arte	11
2.1.	Lepidópteros.....	11
2.1.1.	Diferencias entre mariposas y polillas	12
2.1.2.	Importancia de lepidópteros y su clasificación	13
2.2.	Aprendizaje profundo	14
2.2.1.	Definición.....	15
2.2.2.	Redes neuronales convolucionales	17
3.	Materiales y métodos	19
3.1.	Preprocesamiento y generación de datos	19
3.1.1.	Conjunto de datos	19
3.1.2.	Generación de datos	21
3.2.	Estructura general de entrenamiento	22
3.3.	Modelo secuencial.....	23
3.3.1.	Configuración analizada	23
3.3.2.	Entrenamiento	24
3.3.3.	Evaluación y predicción	27
3.4.	Modelo preentrenado ResNet	30
3.4.1.	Configuraciones analizadas	30
3.4.2.	Entrenamiento	31
3.4.3.	Evaluación y predicción	34
3.5.	Modelo preentrenado EfficientNet	36
3.5.1.	Configuraciones analizadas	36
3.5.2.	Entrenamiento	37
3.5.3.	Evaluación y predicción	40
4.	Resultados	43
5.	Conclusiones y trabajos futuros	45
5.1.	Conclusiones	45
5.2.	Trabajos futuros	46
6.	Glosario.....	47
7.	Bibliografía	49
8.	Anexos	51

Lista de figuras

Figura 1: Diagrama de Gantt	7
Figura 2: Mariposa monarca. Fuente: Wikipedia	11
Figura 3: Polilla.....	12
Figura 4: Mariposa	12
Figura 5: Arquitectura de una red neuronal profunda.....	16
Figura 6: Arquitectura red neuronal convolucional para clasificación.....	17
Figura 7: Imágenes mariposas y polillas dataset Kaggle. Fuente: Kaggle	19
Figura 8: Gráfica distribución dataset entrenamiento.....	20
Figura 9: Arquitectura secuencial	24
Figura 10: Categorical accuracy modelo secuencial.	25
Figura 11: Pérdida modelo secuencial.	25
Figura 12: Precision modelo secuencial. Fuente: Cuaderno Colab	26
Figura 13: Recall modelo secuencial. Fuente: Cuaderno Colab.....	26
Figura 14: F1-score modelo secuencial.	26
Figura 15: Matriz de confusión modelo secuencial. Fuente: Cuaderno Colab .	28
Figura 16: Arquitectura con modelo ResNet.....	30
Figura 17: Categorical accuracy modelo ResNet101V2.....	32
Figura 18: Pérdida modelo ResNet101V2.....	32
Figura 19: Precision modelo ResNet101V2. Fuente: Cuaderno Colab.	33
Figura 20: Recall modelo ResNet101V2. Fuente: Cuaderno Colab	33
Figura 21: F1-score modelo ResNet101V2. Fuente: Cuaderno Colab	33
Figura 22: Matriz de confusión ResNet. Fuente:Cuaderno Colab	35
Figura 23: Arquitectura con modelo EfficientNet	36
Figura 24: Categorical accuracy modelo EfficientNetB5.	38
Figura 25: Pérdida modelo EfficientNetB5. Fuente: Cuaderno Colab	38
Figura 26: Precision modelo EfficientNetB5. Fuente: Cuaderno Colab	39
Figura 27: Recall modelo EfficientNetB5. Fuente: Cuaderno Colab.....	39
Figura 28: F1-score modelo EfficientNetB5. Fuente: Cuaderno Colab.....	39
Figura 29:Matriz de confusión EfficientNetB5. Fuente: Cuaderno Colab	41

1. Introducció

1.1. Contexto y justificación del Trabajo

Las mariposas y las polillas son insectos que forman parte de la orden Lepidóptera (lepidópteros). Se caracterizan por disponer en su mayoría de dos pares de alas con escamas microscópicas en su etapa adulta. Aproximadamente hay 180.000 especies de lepidópteros, ocupando estos insectos el segundo lugar tras los coleópteros (escarabajos) [1].

Los lepidópteros además tienen un valor crucial en el ecosistema. Son importantes en la cadena alimenticia, ya que transforman material vegetal en animal y a su vez son alimento de otros animales. También son insectos polinizadores, función imprescindible para mantener la biodiversidad de las plantas, ya que sobre un 80% de las plantas dependen de la polinización animal.

Existen especies beneficiosas para la economía (seda o alimento). También desempeñan un papel importante en la investigación biológica. Su contribución abarca campos como la ecología, biogeografía, sistemática, genética y fisiología. También hay especies que dañan plantas útiles para los humanos, así que de igual forma es necesario poder identificar estas especies para el control de plagas.

Informes recientes indican que hay una disminución drástica en el número de insectos, entre ellos los lepidópteros, pero la forma de evaluar el alcance del fenómeno es deficiente ya que las poblaciones de insectos son difíciles de estudiar y la mayoría de los métodos son laboriosos y requieren mucha mano de obra [4].

En estudios realizados en Europa se ha revelado que las especies de mariposas de praderas han disminuido en casi un 50% desde principios de los años 1990 y tres cuartas partes de las mariposas en el Reino Unido están en declive [5]. Sólo en Estados Unidos, cinco mariposas se han extinguido desde 1950 [5]; otras veintinueve mariposas están catalogadas como en peligro de extinción en todo el país, y seis están catalogadas como amenazadas [5]. También se han evaluado las 800 especies de mariposas conocidas en los Estados Unidos y se ha descubierto que el 19 % está actualmente en riesgo de extinción [5].

De forma tradicional, la identificación de especies de mariposas se realiza mediante observación visual por parte de expertos, lo cual es una tarea laboriosa y propensa a errores y requiere un conocimiento profesional profundo [2]. Otra forma de clasificar las mariposas es utilizando las claves taxonómicas y la secuenciación de ADN, pero igualmente es un procesamiento manual y debe ser realizada por profesionales cualificados [6].

Esta identificación es bastante compleja ya que existen muchas especies y con gran parecido entre ellas, con lo cual hace que sea un método lento, complicado y costoso. Por estos motivos cada vez es más necesaria la automatización del proceso de identificación para poder obtener resultados de forma óptima, sin la necesidad de disponer de tanta mano de obra. La clasificación precisa de las especies de mariposas tiene una gran importancia práctica para poder salvar especies en peligro de extinción y en el estudio de la taxonomía y la biodiversidad [2].

La identificación automática de insectos a partir de imágenes está ganando interés y popularidad día a día. Ha pasado más de una década desde que los investigadores llevaron a cabo el reconocimiento de especies para conservar aquellas en peligro de extinción [2]. En los últimos tiempos, la detección y el reconocimiento automático de especies de mariposas han llamado la atención de los investigadores, por lo que se han realizado diversas investigaciones sobre el proceso de detección y reconocimiento de mariposas [2].

En estudios consultados se ha empleado métodos tradicionales de Machine Learning, Artificial Neural Networks (ANN), Extreme Learning Machine (ELM) y Deep Learning. Uno de los métodos más utilizados en los últimos tiempos son las redes neuronales convolucionales (CNN) que son modelos de aprendizaje profundo diseñados para trabajar con imágenes. Extraen características de las imágenes para aprender a diferenciarlas. Se pueden desarrollar sistemas de alto rendimiento para la detección, enumeración, clasificación y descubrimiento de especies [2].

En este estudio se investigan modelos de aprendizaje automático basado en Deep Learning con redes neuronales convolucionales para la tarea de clasificación de especies de mariposas y polillas y mejorar la precisión. Para ello se hará uso de un conjunto de datos de Kaggle que consta de 13594 imágenes de 100 especies de mariposas y polillas [7].

1.2. Objetivos del Trabajo

El objetivo principal es el desarrollo de modelos de aprendizaje profundo utilizando redes neuronales convolucionales para la clasificación precisa de diversas especies de mariposas y polillas a partir de sus imágenes. Este proyecto busca explorar y aplicar las técnicas de inteligencia artificial en el campo de la biología y la conservación de la naturaleza, con un enfoque particular en los lepidópteros.

Para alcanzar este objetivo principal, se han establecido varios objetivos específicos que garantizarán un enfoque integral y detallado del proyecto:

- **Revisión de Literatura sobre Lepidópteros:** Se llevará a cabo una revisión detallada de la literatura relacionada con las mariposas diurnas y las polillas, con el fin de obtener un conocimiento profundo sobre estas especies.

- **Análisis de Literatura en Inteligencia Artificial y Aprendizaje Profundo:** Se investigarán los desarrollos actuales en el campo de la inteligencia artificial, centrándose especialmente en las metodologías de aprendizaje profundo aplicadas a la clasificación de imágenes.
- **Evaluación de Algoritmos de Redes Neuronales:** Se investigarán diferentes algoritmos de redes neuronales para seleccionar el más adecuado para la clasificación de imágenes.
- **Selección y Preprocesamiento del Dataset:** Se buscará y evaluará un conjunto de datos adecuado de imágenes de mariposas y polillas, realizando el preprocesamiento necesario para asegurar su calidad y utilidad en el entrenamiento del modelo.
- **Configuración del Entorno de Desarrollo:** Se identificarán las herramientas de programación, librerías y recursos necesarios para el desarrollo del modelo.
- **Desarrollo de Código para Procesamiento de Imágenes y Clasificación:** Se desarrollará el código necesario para el tratamiento de imágenes, la clasificación y el análisis de resultados.
- **Validación de los Resultados:** Se realizará una evaluación rigurosa de los resultados obtenidos para validar la eficacia del modelo.
- **Documentación Completa del Proyecto:** Se preparará una memoria detallada y una presentación del proyecto, que incluirá todas las fases del desarrollo, desde la conceptualización hasta los resultados finales y su análisis.
- **Integración de Técnicas de Mejora y Optimización:** Se explorarán métodos para optimizar el rendimiento del modelo, como ajuste fino, aumento de datos y regularización.
- **Comparación entre Modelos:** Se compararán diversos modelos desarrollados para poder evaluar cual es el más eficaz para este caso concreto.

1.3. Impacto en sostenibilidad, ético-social y de diversidad

La competencia de compromiso ético y global (CCEG) está definida a nivel de Máster como:

Actuar de manera honesta, ética, sostenible, responsable y respetuosa con los derechos humanos y la diversidad, tanto en la práctica académica como en la profesional, y diseñar soluciones para mejorar estas prácticas.

El impacto de este trabajo en las tres dimensiones de la CCEG es el siguiente:

- Sostenibilidad
 - ODS 7 – Energía Asequible y No Contaminante: El uso de redes neuronales puede ser más eficiente energéticamente en comparación con los métodos manuales, promoviendo el uso de tecnologías con menor impacto ambiental.
 - ODS 9 – Industria, Innovación e Infraestructura: La integración de tecnologías avanzadas como las redes neuronales en la clasificación de lepidópteros impulsa la innovación y mejora la infraestructura tecnológica en el campo de la investigación biológica.
 - ODS 11 – Ciudades y Comunidades Sostenibles: La investigación contribuye a la sostenibilidad ambiental urbana al mejorar la comprensión de la biodiversidad y su relación con los entornos urbanos.
- Comportamiento Ético y Responsabilidad Social (RS)
 - ODS 4 – Educación de Calidad: El uso de inteligencia artificial para la clasificación de lepidópteros proporciona un valioso recurso educativo, permitiendo una comprensión más profunda y precisa de la biodiversidad.
 - ODS 17 – Alianzas para Lograr los Objetivos: La colaboración entre investigadores, tecnólogos y educadores en el desarrollo y la aplicación de redes neuronales para la clasificación de lepidópteros fomenta asociaciones efectivas y el intercambio de conocimientos.

- Diversidad (género entre otros) y Derechos Humanos
 - ODS 5 – Igualdad de Género: La democratización del acceso a herramientas de clasificación avanzadas, como las redes neuronales, permite una participación más equitativa en la ciencia, independientemente del género.
 - ODS 10 – Reducción de las Desigualdades: Al facilitar el acceso a tecnologías en países y regiones con menos recursos, se contribuye a reducir la brecha en capacidades científicas y tecnológicas.

1.4. Enfoque y método seguido

En este proyecto de clasificación de imágenes se ha adoptado un método basado en el modelo tradicional en cascada, pero incluyendo iteraciones en etapas clave para adaptarse a las características cambiantes y evolutivas del Deep Learning. Esta estructura combina la organización y claridad del modelo en cascada con la flexibilidad necesaria para experimentar y ajustar el modelo conforme avanza el proyecto. Este enfoque equilibra la metodología estructurada con la capacidad de adaptación, esencial en el campo del Deep Learning.

Se estructura en las siguientes fases:

- Fase de análisis

En esta fase se analiza la selección de los modelos de Deep Learning para la clasificación de imágenes además de la arquitectura del pipeline y las métricas a usar.

En esta fase se comprueba si el dataset cumple todas las características necesarias para su tratamiento o es necesario hacer algún tipo de preprocesado o cambio en los sets de entrenamiento, validación y test.
- Fase de implementación

En esta fase, el desarrollo se lleva a cabo en Python, utilizando Google Colab y las librerías necesarias del modelo seleccionado. Se emplea Keras, una biblioteca de Python de alto nivel para el desarrollo y entrenamiento de modelos de Deep Learning. Keras es conocida por su simplicidad y flexibilidad, siendo una herramienta muy utilizada para construir redes neuronales convolucionales y otros modelos avanzados de aprendizaje automático.
- Fase de entrenamiento y testeo

En esta fase el modelo será entrenado y ajustado de forma iterativa usando los datos de entrenamiento para mejorar su capacidad de

clasificación de imágenes. Se experimentará con diferentes configuraciones para optimizar el rendimiento. También se realizarán pruebas con un conjunto de datos de prueba para asegurarse de que el modelo no solo aprende bien, sino que también puede aplicar lo aprendido a nuevas imágenes. Este proceso iterativo de entrenamiento y prueba ayudará a encontrar el equilibrio adecuado entre precisión y capacidad de generalización de los modelos entrenados.

- Fase de estudio de los resultados
En esta fase se realiza un análisis detallado de los resultados obtenidos de los diferentes modelos que han sido entrenados. El objetivo es determinar cuál de estos modelos muestra un mejor desempeño en la tarea de clasificación de imágenes. Se evaluarán y compararán diversas métricas y la capacidad de generalización, para identificar el modelo más eficaz.

1.5. Planificación del Trabajo

En la siguiente tabla se muestran las distintas tareas de la planificación con sus fechas de inicio y fin, además de indicar los hitos que se corresponden con la entrega de cada PEC.

Tarea	Fecha inicio	Fecha fin	Hito
Definición y plan de trabajo	27-09-2023	16-10-2023	PEC 1
Análisis inicial del contexto	27-09-2023	10-10-2023	
Elaboración de planificación y memoria	05-10-2023	15-10-2023	
Definición del trabajo – Fase 1	17-10-2023	20-11-2023	PEC 2
Análisis literatura biológica	17-10-2023	24-10-2023	
Aprendizaje de machine learning y Deep learning	17-10-2023	20-11-2023	
Investigación entornos ejecución y librerías	20-10-2023	10-11-2023	
Implementación pipeline	26-10-2023	17-11-2023	
Documentación (informe PEC 2 y memoria)	17-10-2023	20-11-2023	
Definición del trabajo – Fase 2	21-11-2023	23-12-2023	PEC 3
Entrenamiento	21-11-2023	20-12-2023	
Test/Métricas	21-11-2023	20-12-2023	
Evaluación de resultados	10-12-2023	23-12-2023	
Documentación (informe PEC 3 y memoria)	21-11-2023	23-12-2023	
Cierre memoria y presentación	27-12-2023	14-01-2024	PEC 4
Redacción final memoria	27-12-2023	07-01-2024	

Elaboración de presentación	27-12-2023	14-01-2024
Defensa pública	22-01-2024	02-02-2024
		PEC 5

Calendario

Se puede observar el diagrama de Gantt en la Figura 1

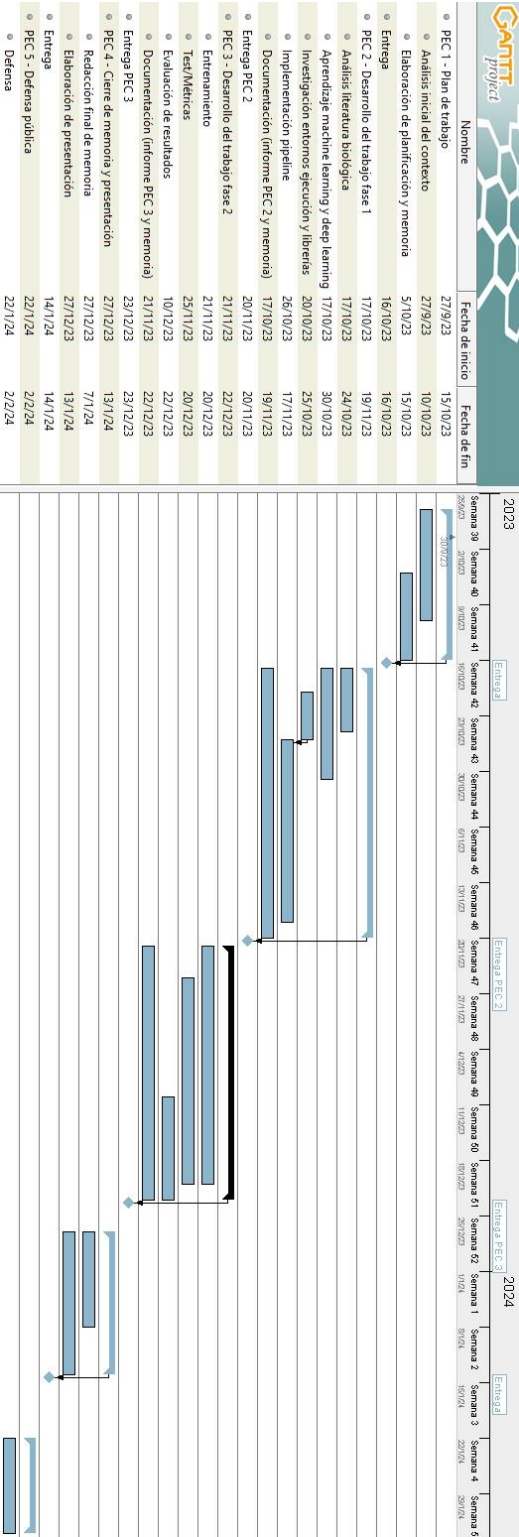


Figura 1: Diagrama de Gantt

Hitos

Los hitos se corresponden con cada una de las entregas parciales:

- PEC 1 – Plan de trabajo 16/10/2023
- PEC 2 – Desarrollo del trabajo fase 1 20/11/2023
- PEC 3 – Desarrollo del trabajo fase 2 23/12/2023
- PEC 4 – Cierre de memoria y presentación 14/01/2024
- PEC 5 – Defensa pública 02/02/2024

Riesgos asociados al TFM

Incumplimiento de fechas en tareas	
Descripción	Por motivos laborales o personales hay posibilidad de que algunas fechas se puedan ver comprometidas.
Severidad	Alta
Probabilidad	Media
Mitigación	Para mitigar este riesgo: <ul style="list-style-type: none">• Disposición de flexibilidad de los días de vacaciones para situarlos en las fechas necesarias.• Tener en cuenta en la planificación que los días de mayor dedicación son viernes, sábado y domingo.• Dejar un margen extra en las tareas para poder reaccionar a este tipo de imprevisto.

No conseguir un resultado satisfactorio	
Descripción	No conseguir los resultados esperados al evaluar el modelo entrenado.
Severidad	Media
Probabilidad	Media
Mitigación	<ul style="list-style-type: none">• Modificación de parámetros y volver a iterar• Modificación de arquitectura del modelo.• Aumento de datos.• Regularización.• Balanceo de datos para prevenir sobreajuste.• Investigar la posibilidad del uso de otro modelo.

Nuevas tareas no planificadas	
Descripción	A medida que el proyecto avance hay posibilidad de que surja alguna tarea no planificada.
Severidad	Media
Probabilidad	Baja
Mitigación	<ul style="list-style-type: none"> • Disposición de un margen para cada fase de entrega para poder encajar cualquier imprevisto.

Pérdida o rotura de archivos resultantes	
Descripción	Pérdida o rotura de los entregables del proyecto, como la memoria, la presentación o el código.
Severidad	Alta
Probabilidad	Baja
Mitigación	<ul style="list-style-type: none"> • Tener siempre una copia de seguridad de cada entregable y además tenerlos también subidos en la nube. • Tener versionado en el caso de código software por evitar modificaciones que lleven a estropear el código.

1.6. Breve resumen de productos obtenidos

El producto final consta de:

- Memoria en formato pdf donde se detalla el proyecto.
- Presentación en formato pdf.
- Presentación virtual.
- Cuaderno Colab con el trabajo realizado. Se puede acceder mediante el enlace indicado en el [Anexo](#).

1.7. Breve descripción de los otros capítulos de la memoria

La organización de este proyecto se presenta a través de una serie de capítulos. A continuación, se ofrece una visión general de cada capítulo, detallando su enfoque y contribuciones específicas:

- Capítulo 2. Estado del arte: Se hace una breve descripción de los lepidópteros, al cual pertenecen las mariposas y polillas. Se describen las características principales de estos insectos y la importancia de su

clasificación. Se revisan las técnicas usadas para la clasificación de imágenes.

- Capítulo 3. Materiales y métodos: En este capítulo se describe el conjunto de datos selecciona y metodología usada en el proyecto. Se describe el conjunto de datos y su preprocesamiento, además de la descripción de los diferentes modelos seleccionados y su entrenamiento y evaluación.
- Capítulo 4. Resultados: En este capítulo se compararán los resultados obtenidos en los modelos seleccionados.
- Capítulo 5. Conclusiones y trabajos futuros.
- Capítulo 6. Glosario

2. Estado del arte

2.1. Lepidópteros

Dentro de la orden Lepidóptera, mariposas (Figura 2) y polillas, encontramos una gran diversidad y una presencia global extensa. Estos insectos pasan por un proceso de metamorfosis completa, transformándose desde huevos a larvas (orugas), luego a pupas (crisálidas), y finalmente emergiendo como adultos. Un rasgo distintivo de esta orden es la presencia de escamas en sus alas, que son responsables de los variados y atractivos patrones de colores. De hecho, el nombre "Lepidóptera" se deriva del griego y refleja esta característica única: "lepidos" significa escama y "pteron", ala.



Figura 2: Mariposa monarca. Fuente: Wikipedia

Con más de 180,000 especies identificadas [1], los lepidópteros se adaptan a una amplia variedad de ambientes, desde los desiertos más secos hasta las selvas tropicales más húmedas. Su capacidad para prosperar en diversos ecosistemas habla de su extraordinario éxito evolutivo. Estos insectos no solo son interesantes en términos de diversidad y adaptabilidad, sino que también desempeñan roles ecológicos de vital importancia. Muchas mariposas, por ejemplo, son polinizadoras esenciales para diversas plantas florales. De forma similar, diversas polillas desempeñan un papel crucial en la polinización nocturna, lo que es fundamental para el equilibrio de ciertos ecosistemas y la propagación de varias especies de plantas.

La variedad de comportamientos y adaptaciones en los lepidópteros, como las mariposas y las polillas, es realmente asombrosa y juega un papel crucial en sus ecosistemas. Por ejemplo, la mariposa monarca es bien conocida por sus

largas migraciones. Las polillas, por su parte, también muestran adaptaciones notables, particularmente en el ámbito del camuflaje. Un caso destacado es la polilla de la corteza del abedul (*Endromis versicolora*), que se camufla hábilmente para parecerse a la corteza de los árboles, una estrategia de evasión efectiva contra los depredadores. Tanto las mariposas como las polillas, con sus variadas tácticas de mimetismo y camuflaje, como el mimetismo Batesiano y Mülleriano, demuestran la extraordinaria complejidad de la evolución y su impacto en la supervivencia y dinámica de sus respectivos hábitats.

2.1.1. Diferencias entre mariposas y polillas

Las mariposas (Figura 4) y polillas (Figura 3) presentan distintas adaptaciones a sus respectivos entornos. Las características descritas a continuación son generalizaciones comunes, la diversidad entre la gran cantidad de especies en cada grupo hace que haya numerosas excepciones. Las principales diferencias se enumeran a continuación:



Figura 3. Polilla
Fuente: Dataset Kaggle



Figura 4. Mariposa
Fuente: Dataset Kaggle

- **Horario de actividad:** Durante el día, las mariposas se activan y aprovechan la luz solar para sus actividades de alimentación y reproducción. En contraste, las polillas prefieren la nocturnidad, donde eluden a sus depredadores y se benefician de un ambiente más fresco. No obstante, hay especies de polillas que también son activas durante el día.
- **Aspecto físico y tamaño:** Las mariposas suelen ser de mayor tamaño, ostentando colores vivos y diseños llamativos en sus alas, útiles tanto en el cortejo como en la defensa contra depredadores. Por otro lado, las polillas, generalmente más pequeñas y de aspecto menos vistoso, presentan colores y patrones más sutiles, ideales para el camuflaje en su hábitat nocturno y así aumentar sus posibilidades de supervivencia.

- Estructura corporal: Las mariposas tienen un cuerpo esbelto y diseñado para un vuelo eficaz durante las horas del día. Las polillas, en cambio, presentan una estructura corporal más densa y peluda, lo que les sirve como aislamiento térmico durante las frías noches.
- Diseño de antenas: Las mariposas se caracterizan por antenas finas y estilizadas con un extremo bulboso, una adaptación que mejora su habilidad sensorial y comunicativa. En cambio, las polillas muestran una amplia gama de formas en sus antenas, desde las plumosas hasta las lineales sin bulbo, lo que les confiere una mayor capacidad para detectar olores en el aire.
- Posición de las alas en reposo: Cuando descansan, las mariposas cierran sus alas hacia arriba, reduciendo su visibilidad como medida de camuflaje. Por el contrario, las polillas optan por descansar con las alas abiertas, lo que les permite integrarse con su entorno y pasar inadvertidas.
- Ciclo de Vida: Tanto las mariposas como las polillas experimentan un ciclo de vida similar, con etapas de huevo, larva, pupa y adulto. Sin embargo, se diferencian en sus preferencias de hábitat y comportamiento en cada etapa. Las mariposas tienden a estar más ligadas a plantas y entornos específicos, mientras que las polillas muestran una adaptabilidad más amplia a diversos ambientes.

2.1.2. Importancia de lepidópteros y su clasificación

Estos insectos juegan un papel multifacético en la ecología. Como polinizadores, son fundamentales para la salud y proliferación de una amplia gama de plantas. Al alimentarse, facilitan la transferencia de polen entre las flores, un proceso vital para la reproducción de muchas especies vegetales y para la producción de alimentos. Además, estos insectos sirven como indicadores clave de la salud ambiental, reflejando cambios ecológicos y señalando problemas como la pérdida de hábitat o la contaminación.

En la cadena alimentaria, las orugas de los lepidópteros son un componente esencial en la dieta de muchas aves y mamíferos pequeños. Su presencia en un ecosistema es un indicador importante de la salud de las redes tróficas y contribuye significativamente al equilibrio natural. La diversidad genética y de especies de estos insectos también enriquece la biodiversidad global, aportando al equilibrio y estabilidad de los ecosistemas.

Por último, mariposas y polillas han sido objeto de extensos estudios científicos que han contribuido a una mayor comprensión de procesos ecológicos y evolutivos. Estas investigaciones han aportado conocimientos valiosos en

áreas como la genética y la biología evolutiva, expandiendo el entendimiento del mundo natural y de la complejidad de la vida en la Tierra.

La clasificación de los lepidópteros es fundamental en una gran variedad de campos, tanto desde una perspectiva científica como ecológica:

- **Rol en Biodiversidad y Ecología:** Los lepidópteros son vitales para evaluar la biodiversidad y el estado de los ecosistemas. Su clasificación proporciona información que ayuda a comprender la distribución y diversidad de especies en distintos ecosistemas. Además, desempeñan un rol vital tanto como polinizadores y como parte de la cadena alimentaria.
- **Conservación:** La clasificación precisa es esencial para la conservación de especies. Al identificar y categorizar especies amenazadas o en peligro, los expertos pueden desarrollar estrategias de protección más eficientes.
- **Estudios Evolutivos y Genéticos:** La clasificación de lepidópteros permite a los investigadores analizar las tendencias evolutivas y las relaciones genéticas entre las distintas especies. Este análisis ofrece información valiosa sobre su adaptación a distintos entornos y cómo podrían reaccionar ante futuros cambios en sus ecosistemas.
- **Investigación Agrícola:** Algunas especies de polillas son plagas agrícolas significativas. Comprender su clasificación es de vital importancia para desarrollar métodos de control de plagas eficaces y a su vez sostenibles.
- **Impulso en Biomedicina y Biotecnología:** La investigación en lepidópteros contribuye significativamente a la genética, el desarrollo de nuevos materiales y las ciencias médicas. Un área de interés particular es el estudio de la seda producida por ciertas polillas, investigada por su aplicabilidad en el campo biomédico.

2.2. Aprendizaje profundo

En el contexto del machine learning (ML), el aprendizaje profundo o deep learning se ha establecido como un avance significativo en la inteligencia artificial, caracterizado por el uso de redes neuronales artificiales con diversas capas. Estas redes, diseñadas para emular el funcionamiento del cerebro humano, son excepcionales en su capacidad para aprender y procesar una amplia gama de datos. La estructura multicapa de estas redes facilita una exploración detallada y profunda de datos complejos, convirtiéndolas en

herramientas ideales para tareas sofisticadas como la detección de patrones en el lenguaje, reconocimiento de voz y análisis de imágenes.

En lo que respecta a la clasificación de imágenes, el aprendizaje profundo ha traído una revolución, logrando niveles de precisión y eficiencia extraordinarios. Las redes neuronales convolucionales, especializadas en reconocer patrones visuales de imágenes, han traído avances significativos en campos como en el diagnóstico médico, en seguridad y análisis geográfico.

En el campo de la biología, la influencia del aprendizaje profundo en la clasificación de especies es muy significativa. Permite a los expertos analizar extensos volúmenes de datos, como imágenes de especies o secuencias genéticas, con una precisión y rapidez imposible de alcanzar manualmente. Esto es fundamental para descubrir nuevas especies, monitorear la biodiversidad y comprender la evolución.

En la entomología y la conservación de lepidópteros, el uso del aprendizaje profundo, en particular de redes neuronales convolucionales, es fundamental en la clasificación de imágenes. Estas redes, al analizar fotografías y datos genéticos, pueden identificar especies de mariposas y polillas con gran precisión. Este nivel de detalle es vital para la conservación de la biodiversidad, mejorando la capacidad de los científicos para investigar migraciones y evoluciones de estas especies y proporcionando datos indispensables para la protección de su diversidad biológica y sus ecosistemas.

2.2.1. Definición

El Aprendizaje Profundo (Deep Learning), que es una subdivisión del Aprendizaje Automático (Machine Learning), utiliza redes neuronales artificiales diseñadas inspirándose en el funcionamiento neurológico del cerebro humano. Estas redes neuronales se caracterizan por su estructura compuesta de múltiples capas de nodos o neuronas, las cuales están interconectadas entre sí para procesar y transmitir señales de manera secuencial y jerárquica.

Cada neurona en estas redes funciona como una unidad de procesamiento que recibe, procesa y transmite información. La eficacia de estas redes está en su habilidad para realizar cálculos complejos a través de la combinación y transformación de los datos de entrada en cada capa sucesiva. A medida que los datos avanzan a través de las capas de la red, se lleva a cabo una extracción cada vez más refinada de características, permitiendo que la red identifique y aprenda patrones y relaciones.

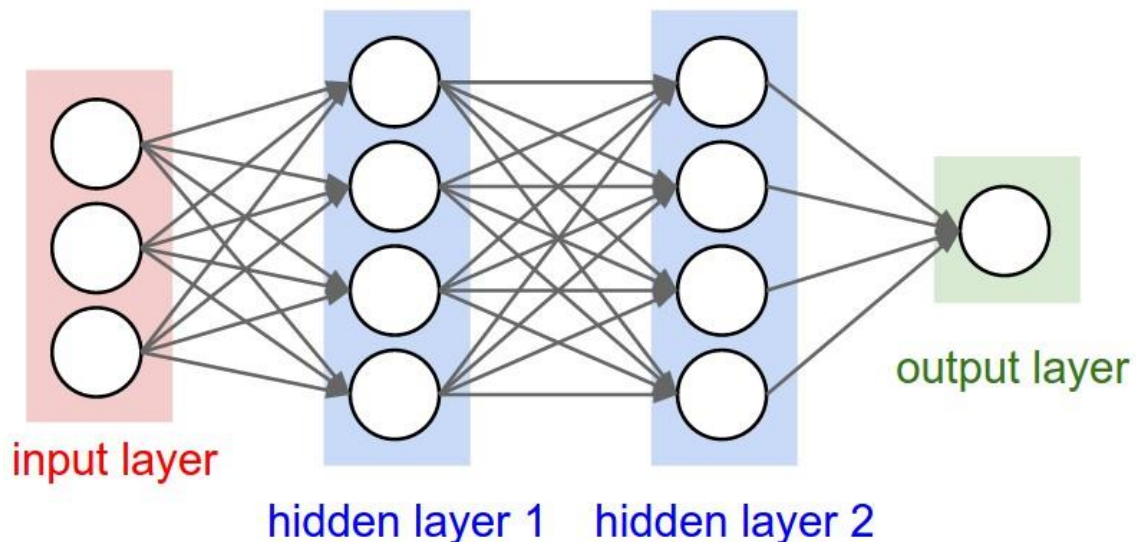


Figura 5: Arquitectura de una red neuronal profunda.
 Autor desconocido bajo licencia [CC BY-SA](#)

La arquitectura básica de una red neuronal profunda (ver Figura 5) consta de las siguientes capas:

- Capa de Entrada (Input Layer): Es la capa de entrada de los datos a procesar en la red neuronal.
- Capas Ocultas (Hidden Layers): Son las capas responsables de la mayor parte del procesamiento. Estas capas reciben datos de la capa anterior, los cuales se transforman mediante una combinación de pesos y sesgos, seguidos de la aplicación de funciones de activación no lineales como ReLU, sigmoide o tanh. La profundidad y el tamaño de estas capas varían según la complejidad de la tarea, permitiendo a la red aprender y extraer características y patrones cada vez más complejos de los datos.
- Capa de Salida (Output Layer): Esta capa procesa el resultado final. En tareas de clasificación, se emplea comúnmente una función de activación como softmax, que clasifica las entradas en las distintas categorías de la clasificación.

Durante el entrenamiento de estas redes, los pesos y sesgos se ajustan a través de un proceso conocido como retropropagación, utilizando algoritmos como el Descenso de Gradiente.

En el ámbito del aprendizaje supervisado, particularmente en tareas de clasificación como es el caso de este proyecto, las redes neuronales de aprendizaje profundo se entrenan con conjuntos de datos etiquetados. Cada entrada en estos conjuntos tiene una etiqueta o categoría asignada, permitiendo a la red aprender a clasificar nuevas entradas de manera precisa. Con cada ejemplo procesado, la red mejora su capacidad de clasificación, convirtiéndose en una herramienta poderosa para tareas de reconocimiento y clasificación.

2.2.2. Redes neuronales convolucionales

Las Redes Neuronales Convolucionales (CNN) son un tipo específico de redes neuronales especialmente diseñadas para procesar datos con estructura matricial, como es el caso de las imágenes, lo que las hace muy útiles en aplicaciones relacionadas con la visión por computador y el procesamiento de imágenes. Estas redes automatizan y perfeccionan la extracción de características, una tarea que en los sistemas de visión artificial tradicionales requieren un diseño técnico detallado y gran experiencia. Gracias a su capacidad para manejar grandes volúmenes de datos y su gran potencia de procesamiento, las CNN pueden aprender a reconocer una diversidad de objetos en imágenes con una muy buena precisión. Esto las convierte en herramientas ideales para una variedad de aplicaciones prácticas, como el reconocimiento facial, la detección de objetos o la clasificación de imágenes.

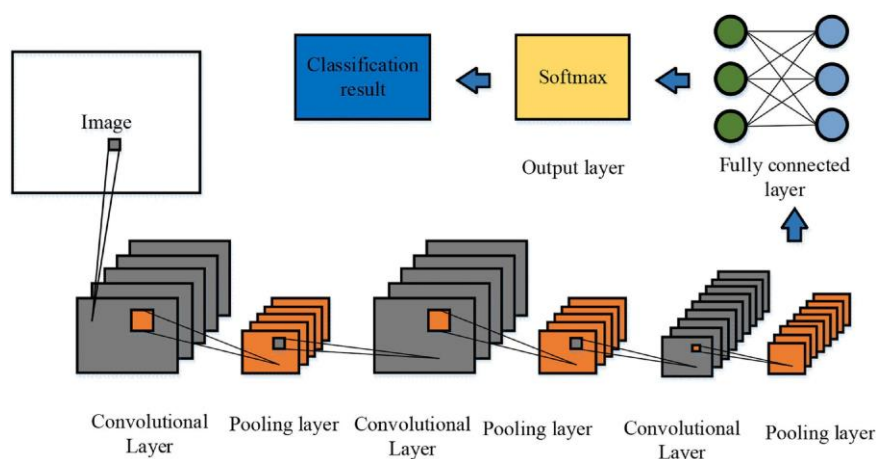


Figura 6: Arquitectura red neuronal convolucional para clasificación.

Fuente: Autor desconocido bajo licencia [CC BY](#)

Una CNN consta de diversas capas con una arquitectura similar a la mostrada en la Figura 6. La funcionalidad de cada una de las capas es la siguiente:

- **Capas Convolucionales:** Son el elemento central de una CNN. Emplean filtros para ejecutar operaciones matemáticas, facilitando la detección de patrones como bordes, texturas y otros aspectos visuales en las imágenes. A medida que los datos avanzan por sucesivas capas convolucionales, la red adquiere la capacidad de reconocer patrones de mayor complejidad.
- **Capas de Activación:** Generalmente situada después de las capas convolucionales, estas capas, mediante funciones de activación como por ejemplo ReLU, introducen no linealidades en el modelo. Esto permite a la red aprender una gama más amplia de representaciones.

- Capas de Agrupamiento (Pooling): Tienen la función de disminuir las dimensiones de los datos de entrada, reduciendo la cantidad de parámetros y el cálculo necesario en la red. Una de las operaciones más comunes es max pooling, que selecciona el valor máximo de un grupo de píxeles, simplificando así la información que se transfiere a la capa siguiente.
- Capas Completamente Conectadas (Fully Connected): Situadas en la parte final de la red. Estas capas integran todas las características identificadas por las capas previas para efectuar la clasificación final.
- Capa de Salida: Es la última capa de la red donde se obtiene la predicción final. En tareas de clasificación, normalmente consiste en un conjunto de valores que indican las probabilidades de pertenencia de la entrada a cada clase posible. En tareas de clasificación multiclase, esta capa a menudo implementa una función softmax, que asigna una probabilidad a cada clase posible, asegurando que la suma total de estas probabilidades sea igual a uno. La clase con la probabilidad más alta se considera como la predicción de la red para la entrada dada.

3. Materiales y métodos

3.1. Preprocesamiento y generación de datos

3.1.1. Conjunto de datos

El conjunto de datos sobre el que se realiza el proyecto contiene 13594 imágenes a color de 100 especies de mariposas y polillas, se han obtenido de la plataforma Kaggle, que es una de las fuentes más populares de datos públicos.

El origen de las imágenes de este conjunto de datos se obtuvo en búsquedas de Google sobre estas 100 especies de mariposas. Las imágenes descargadas se verificaron con un detector de imágenes duplicadas para evitar tener imágenes comunes entre el conjunto de prueba, validación y entrenamiento. Las imágenes se recortaron para que la región de interés (ROI) ocupara al menos el 50 % de los píxeles de la imagen y se les cambió el tamaño a 224 X 224 X3 con formato jpg (Figura 7).

Las imágenes están distribuidas en directorio de entrenamiento, validación y test y en cada uno de ellos también hay 100 directorios por cada una de las especies:

- Conjunto de entrenamiento: 12594 imágenes divididas en 100 clases, uno por cada especie y en cada una de ellas hay más de 100 imágenes.
- Conjunto de validación: 500 imágenes divididas en 100 clases, con 5 imágenes por clase.
- Conjunto de test: 500 imágenes divididas en 100 clases, con 5 imágenes por clase.



Figura 7: Imágenes mariposas y polillas dataset Kaggle. Fuente: Kaggle

Al disponer de un conjunto de datos separado en imágenes de entrenamiento, validación y test, no es necesario realizar divisiones en los distintos subconjuntos.

La distribución del conjunto de datos de entrenamiento (Figura 8) tiene una distribución no uniforme, siendo la categoría BROWN ARGUS la que contiene una mayor cantidad de imágenes, con un total de 169, en contraste con SIXSPOT BURNET MOTH, que tiene la menor cantidad con solo 100 imágenes. Esta variación en el número de imágenes por categoría complica el entrenamiento de redes neuronales convolucionales, que funcionan mejor con conjuntos de datos equilibrados, y puede afectar tanto a la precisión en la evaluación como la fiabilidad de las predicciones. El desequilibrio puede provocar un aprendizaje sesgado dificultando la generalización y la detección precisa de categorías menos comunes, repercutiendo directamente en la calidad de los resultados de evaluación y predicciones.

Número de Muestras por Clase en el Conjunto de Entrenamiento

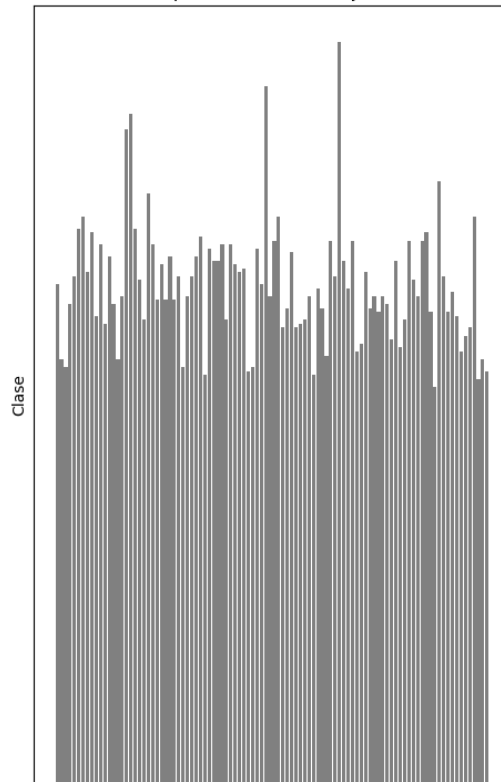


Figura 8: Gráfica distribución dataset entrenamiento.
Fuente: Cuaderno Colab

Para mitigar posibles sesgos a causa del desbalanceo, se aplicarán técnicas de enriquecimiento de datos usando el generador de imágenes de Keras. Este enfoque ampliará la cantidad de imágenes en las categorías menos representadas, además de añadir mayor variabilidad al conjunto de datos, un factor clave para reforzar la capacidad de generalización de la red ante nuevas imágenes no vistas durante el entrenamiento. Además, para ajustar más el balance del conjunto de datos, se calculan y aplican pesos específicos para cada clase mediante la función "compute_class_weight" de la biblioteca "scikit-

learn". Esto asigna mayor importancia a las clases menos representadas durante el proceso de entrenamiento, asegurando que el modelo no se sesgue y mejore su capacidad de reconocer de manera equitativa todas las clases.

La variedad de las imágenes, respecto a ángulos, iluminación y fondos, fortalece la robustez del modelo. Procedentes de búsquedas en Google, estas imágenes reflejan una amplia gama de condiciones, fundamental para el entrenamiento eficiente de una red neuronal convolucional capaz de identificar diferentes especies en una variedad de contextos y situaciones. Además, la ocupación del 50% de las imágenes por la ROI optimiza el entrenamiento al centrar la atención en detalles importantes de las mariposas y polillas, asegurando así una clasificación más efectiva y precisa.

3.1.2. Generación de datos

En el desarrollo de modelos de redes neuronales convolucionales con Keras, los generadores de imágenes son fundamentales para el manejo eficiente y la transformación adecuada de grandes conjuntos de datos de imágenes. Estos generadores optimizan la carga y procesamiento eficiente de imágenes aplicando técnicas de aumento de datos y la normalización para mejorar la precisión y la generalización del modelo.

El generador de entrenamiento implementa diversas técnicas de transformación de imágenes. Estas transformaciones, que se aplican de manera aleatoria durante cada época de entrenamiento, incrementan la diversidad de los datos, lo que contribuye a mejorar el rendimiento del modelo. Las transformaciones realizadas son:

- Normalización de píxeles: Se escalan los valores de los píxeles de un rango de 0-255 a 0-1, optimizando así el rendimiento de las funciones de activación y los optimizadores.
- Cizallamiento aleatorio: Altera las imágenes dentro de un rango específico, añadiendo variabilidad.
- Zoom aleatorio: Modifica el enfoque de las imágenes en un rango del 20%.
- Volteo horizontal aleatorio: Refleja aleatoriamente algunas imágenes horizontalmente, aumentando la diversidad del conjunto.

Además, las imágenes se mantienen en un tamaño estándar de 224x224 y se procesan en lotes de 32, lo que garantiza la consistencia y eficiencia en el manejo de los datos.

Por otro lado, los generadores usados para los conjuntos de validación y prueba adoptan un enfoque más simplificado, centrado únicamente en la normalización de la intensidad de los píxeles de 0 a 1. Esta estrategia garantiza que las imágenes se evalúen en su estado más natural, permitiendo así una evaluación más precisa y realista del rendimiento del modelo en condiciones

que simulan escenarios reales, destacando su capacidad para generalizar a nuevos datos.

3.2. Estructura general de entrenamiento

Tras realizar las primeras pruebas de entrenamiento de modelos y tras encontrar problemas como el sobreajuste o desbalanceo de imágenes, la configuración general del entrenamiento de los modelos usados tiene las siguientes características:

- **Número de Épocas (Epochs):** El número de épocas o veces que se procesa el conjunto de datos completo se ha establecido en 30. Demasiadas épocas pueden llevar a un sobreajuste, mientras que muy pocas pueden resultar en un modelo subajustado.
- **Optimizador y Tasa de Aprendizaje:** Se han realizado experimentos tanto con el optimizador Adam como con SGD, también se han usado distintas tasas de aprendizaje además de las predefinidas por defecto en ambos optimizadores. La tasa de aprendizaje determina el tamaño de los pasos que se toman para alcanzar el mínimo de la función de pérdida.
- **Función de Pérdida:** La función de pérdida seleccionada para todos los experimentos, al tratarse de clasificación multiclase, es "categorical_crossentropy". Esta función mide el rendimiento del modelo en la asignación de la categoría correcta.
- **Métricas de Rendimiento:** Las métricas usadas para obtener información detallada sobre diferentes aspectos del rendimiento del modelo son:
 - **Precisión Categórica (CategoricalAccuracy):** Mide el porcentaje de predicciones correctas del modelo.
 - **Precisión (Precision):** Calcula qué proporción de las predicciones positivas hechas por el modelo son realmente correctas.
 - **Sensibilidad (Recall):** Determina qué proporción de los casos positivos reales fueron identificados correctamente por el modelo.
 - **Puntuación F1 (F1-score):** Refleja el equilibrio entre la precisión y la sensibilidad del modelo. Útil para tener una idea más completa del rendimiento del modelo en tareas de clasificación con desequilibrio de clases.
- **Uso de callbacks para mejorar la eficacia y la eficiencia del entrenamiento del modelo,** ayudando a evitar problemas de sobreajuste y asegurando que el modelo no solo aprenda, sino que también mantenga su capacidad para generalizar a nuevos datos. Estos

callbacks se ejecutan en función del valor de la pérdida en cada época en el conjunto de validación. Se usan:

- ReduceLROnPlateau: Ajusta la tasa de aprendizaje en función del rendimiento, lo que puede ayudar a mejorar la eficacia del modelo. Disminuye un 20% la tasa de aprendizaje si no hay mejora en tres épocas en la pérdida de validación.
- EarlyStopping: Previene el sobreajuste deteniendo el entrenamiento cuando no se observan mejoras. Está configurado para que se detenga el entrenamiento si en 5 épocas no hay mejora en las pérdidas de validación.
- Ponderación de clases: Se ponderan las clases en el entrenamiento para que no haya un desbalanceo, ya que el conjunto de datos es desequilibrado en cuanto a imágenes por clase, algo que se comprobó que afectaba de forma bastante notable en la evaluación del modelo.
- Pasos por Época y Pasos de Validación: Definen el número de lotes de datos procesados por época. El tamaño del lote de imágenes tanto del generador de entrenamiento como el de validación es de 32:
 - Entrenamiento: $\text{Imágenes/tamaño batch} = 12594/32 = 394$
 - Validación: $\text{Imágenes/tamaño batch} = 500/32 = 16$

3.3. Modelo secuencial

Como primera opción seleccionada, se ha optado por un modelo secuencial. Esta arquitectura de red se caracteriza por su estructura lineal, en la cual cada capa se alimenta secuencialmente de la salida de la capa anterior. Esta disposición simplifica tanto la construcción del modelo como la extracción gradual de características importantes, haciéndola ideal para tareas de clasificación. Es especialmente útil en la clasificación de imágenes, donde cada capa se concentra en identificar y aprender patrones cada vez más detallados y específicos, una capacidad esencial para distinguir con eficacia entre diversas categorías.

3.3.1. Configuración analizada

Se realizaron diversos experimentos inicialmente añadiéndose progresivamente más capas de convolución y Max Pooling hasta finalmente llegar al esquema de la Figura 9.

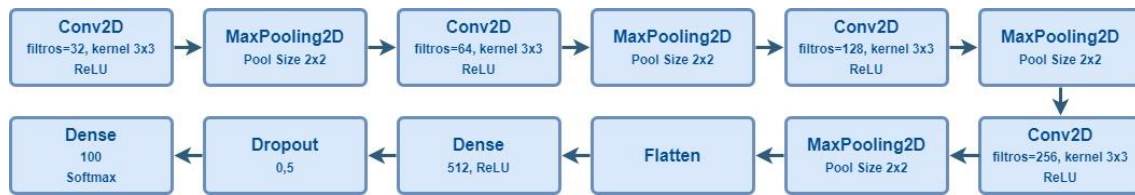


Figura 9. Arquitectura secuencial

La arquitectura consta de las siguientes capas:

- **Bloques Convolucionales:** Son cuatro bloques, cada uno consta de una capa convolucional seguida por una capa de Max Pooling. Las capas convolucionales, con filtros de 32, 64, 128 y 256 y un kernel de 3x3, son responsables de extraer características visuales clave de las imágenes. Esta progresión en el tamaño de los filtros permite capturar características desde las más simples a las más complejas, controlando eficazmente el riesgo de sobreajuste y la gestión de los recursos computacionales. Las capas de Max Pooling, con un tamaño de 2x2, reducen la dimensionalidad de estas características, ayudando a disminuir el número total de parámetros.
- **Flattening:** Esta etapa transforma los mapas de características 2D resultantes de las capas convolucionales en un vector unidimensional. Este paso es necesario para hacer la transición de la extracción de características a la clasificación.
- **Capa Densa:** Incorpora una capa totalmente conectada con 512 unidades y la función de activación ReLU. Esta capa actúa como un clasificador sobre los rasgos extraídos, proporcionando la capacidad de aprender combinaciones complejas de características.
- **Dropout (50%):** Se introduce para reducir el sobreajuste, desconectando aleatoriamente el 50% de las neuronas durante el entrenamiento. Esto evita que el modelo dependa excesivamente de ciertos patrones o características del conjunto de datos de entrenamiento.
- **Capa de Salida:** Se utiliza la función de activación Softmax. Esta capa distribuye la probabilidad de la imagen entre las 100 categorías, para la clasificación multiclase.

3.3.2. Entrenamiento

Tras realizar diversos experimentos modificando el optimizador como la tasa de entrenamiento el mejor resultado se obtiene con el optimizador ADAM con una tasa de aprendizaje de 0.0005.

Del entrenamiento se puede observar:

- Hay mejora en todas las métricas a lo largo de las épocas:

- Precisión Categórica (Figura 10): Hay una mejora constante en la precisión categórica tanto en el conjunto de entrenamiento como en el de validación. Comenzando con 0.1669 y llegando a 0.9223 en el conjunto de entrenamiento y 0.8660 en el de validación.

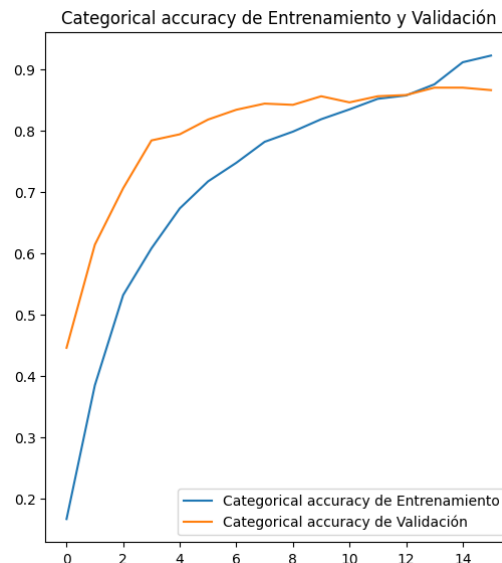


Figura 10. Categorical accuracy modelo secuencial.
Fuente: Cuaderno Colab.

- Pérdida (Loss) (Figura 11): La pérdida disminuye significativamente con cada época, lo cual indica que el modelo está aprendiendo eficientemente.

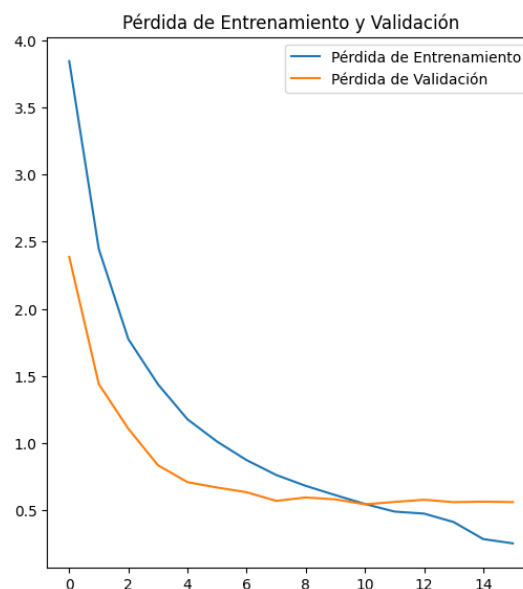


Figura 11: Pérdida modelo secuencial.
Fuente: Cuaderno Colab

- Precision (Figura 12) y Recall (Figura 13): Ambos aumentan consistentemente, lo que sugiere que el modelo está mejorando en identificar las clases correctamente y en cubrir la mayoría de los casos positivos reales.

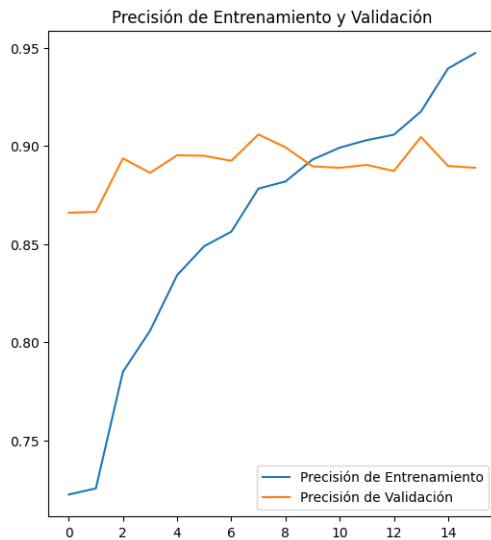


Figura 12: Precision modelo secuencial.
Fuente: Cuaderno Colab

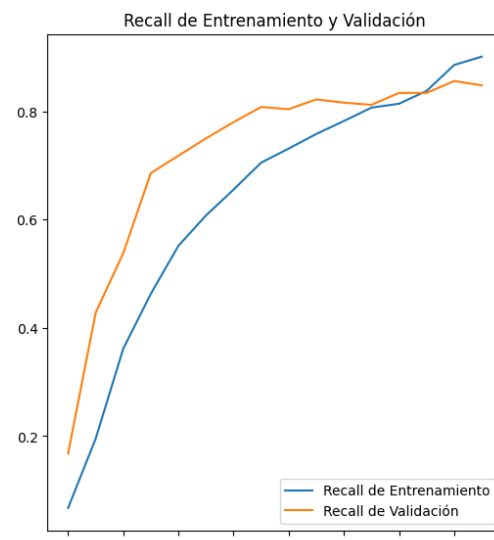


Figura 13: Recall modelo secuencial. Fuente: Cuaderno Colab

- F1-score (Figura 14): El aumento del F1-score en el conjunto de validación (de 0.2814 a 0.8680) indica un equilibrio entre precisión y recall.

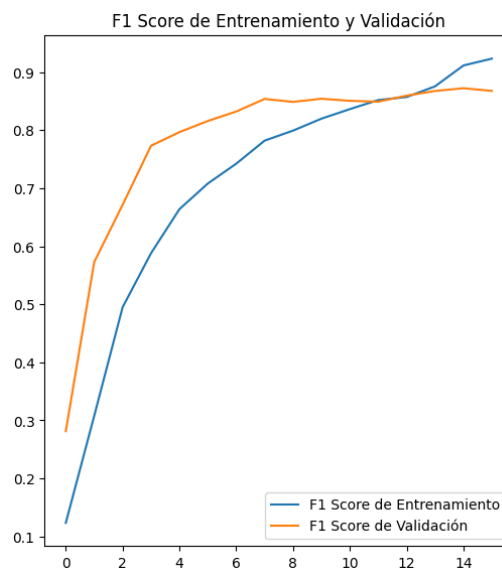


Figura 14: F1-score modelo secuencial.
Fuente: Cuaderno Colab

- Reducción de la Tasa de Aprendizaje en la época 14, momento en el que el modelo deja de mejorar con la tasa actual durante tres épocas consecutivas.
- Detención temprana en la época 16, el modelo dejó de mejorar en el conjunto de validación durante 5 épocas consecutivas, por ello se produce la detención para evitar sobreajuste.

3.3.3. Evaluación y predicción

Tras evaluar el modelo con el conjunto de datos de test se obtienen los siguientes resultados:

Categorical accuracy	0.9079999923706055
Loss	0.35582235455513
Precision	0.9278350472450256
Recall	0.8999999761581421
F1 Score	0.9137054681777954

- Precisión Categórica (Categorical Accuracy): Un valor de 90.80% indica que el modelo es capaz de clasificar correctamente una proporción muy alta. En la mayoría de los casos, el modelo identifica con éxito la categoría correcta.
- Pérdida (Loss): Un valor de 0.3558 es un valor moderado de error, un valor más bajo sería preferible.
- Precisión (Precision): Con un valor de 92.78%, indica que de todas las predicciones de clases positivas un porcentaje bastante elevado fue correcto.
- Recall (Sensibilidad): Con un valor de 90%, el modelo logra detectar la mayoría de las clases positivas en los datos.
- F1-score: Un valor de 91.37% es un equilibrio muy bueno entre precisión y sensibilidad, e indica que el modelo está clasificando bien.

Tras realizar predicción con el conjunto de datos de test se ha obtenido tanto la matriz de confusión (Figura 15) como el informe de clasificación. Se puede observar que 58 de las 100 especies no han conseguido una predicción perfecta.

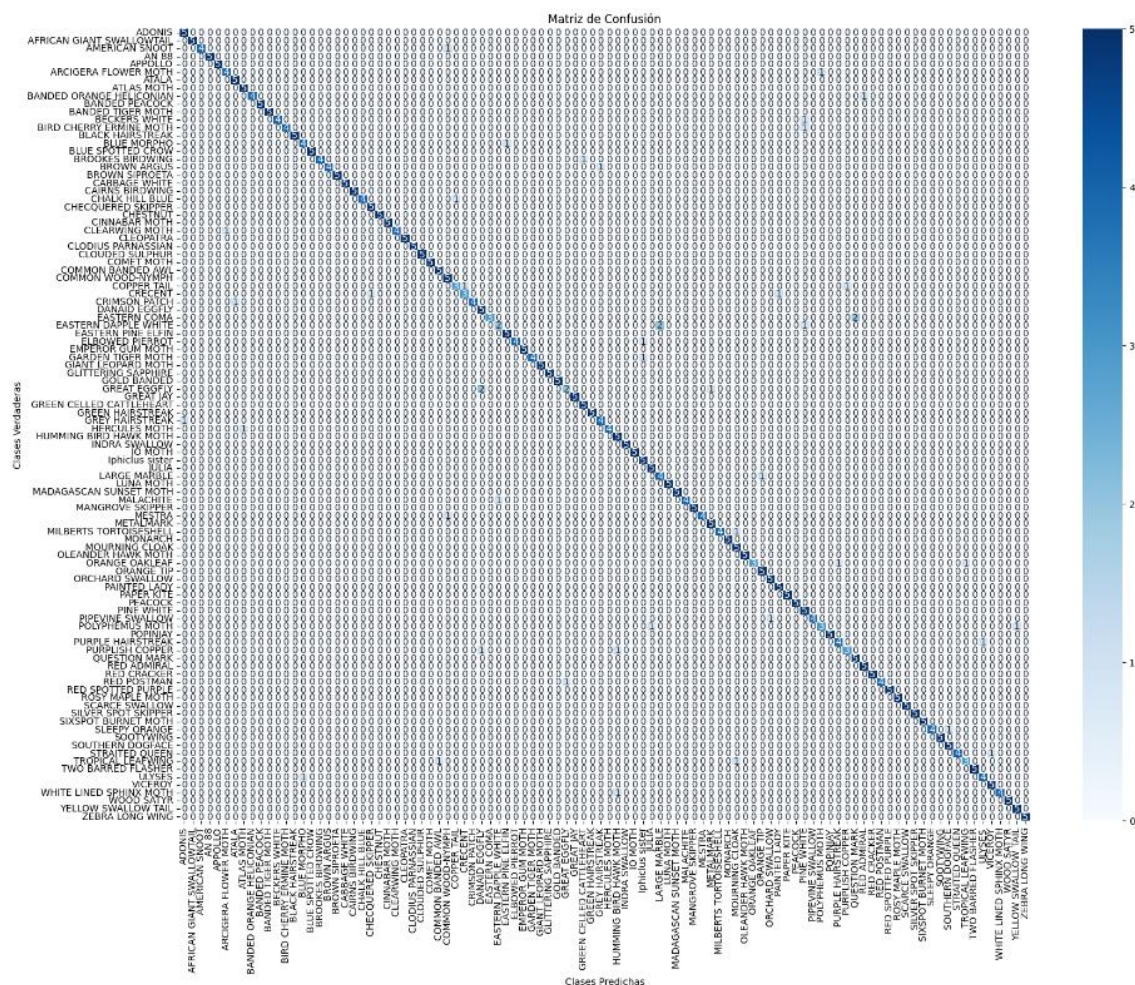


Figura 15: Matriz de confusión modelo secuencial. Fuente: Cuaderno Colab

Especies no categorizadas correctamente en la predicción en todas las imágenes del conjunto de datos de prueba:

Especie	Precision	Recall	F1-score
GREAT EGGFLY	0.67	0.40	0.50
EASTERN DAPPLE WHITE	0.67	0.40	0.50
POLYPHEMUS MOTH	0.75	0.60	0.67
PURPLISH COPPER	0.75	0.60	0.67
COPPER TAIL	0.75	0.60	0.67
CRECENT	0.75	0.60	0.67
TROPICAL LEAFWING	0.75	0.60	0.67
LARGE MARBLE	0.67	0.80	0.73
EASTERN COMA	1.00	0.60	0.75
ORANGE OAKLEAF	1.00	0.60	0.75
DANAID EGGFLY	0.62	1.00	0.77
PINE WHITE	0.62	1.00	0.77
ULYSES	0.80	0.80	0.80
ARCIGERA FLOWER MOTH	0.80	0.80	0.80
BLUE MORPHO	0.80	0.80	0.80

GREY HAIRSTREAK	0.80	0.80	0.80
PURPLE HAIRSTREAK	0.80	0.80	0.80
COMMON WOOD-NYMPH	0.71	1.00	0.83
HUMMING BIRD HAWK MOTH	0.71	1.00	0.83
Iphiclus sister	0.71	1.00	0.83
MOURNING CLOAK	0.71	1.00	0.83
QUESTION MARK	0.71	1.00	0.83
RED POSTMAN	1.00	0.80	0.89
SLEEPY ORANGE	1.00	0.80	0.89
STRAITED QUEEN	1.00	0.80	0.89
WHITE LINED SPHINX MOTH	1.00	0.80	0.89
PIPEVINE SWALLOW	1.00	0.80	0.89
HERCULES MOTH	1.00	0.80	0.89
MALACHITE	1.00	0.80	0.89
MESTRA	1.00	0.80	0.89
AMERICAN SNOOT	1.00	0.80	0.89
BANDED ORANGE HELICONIAN	1.00	0.80	0.89
BECKERS WHITE	1.00	0.80	0.89
BIRD CHERRY ERMINE MOTH	1.00	0.80	0.89
BROOKES BIRDFWING	1.00	0.80	0.89
BROWN ARGUS	1.00	0.80	0.89
CHALK HILL BLUE	1.00	0.80	0.89
CLEARWING MOTH	1.00	0.80	0.89
CRIMSON PATCH	1.00	0.80	0.89
ELBOWED PIERROT	1.00	0.80	0.89
GARDEN TIGER MOTH	1.00	0.80	0.89
MILBERTS TORTOISESHELL	1.00	0.80	0.89
EASTERN PINE ELFIN	0.83	1.00	0.91
RED ADMIRAL	0.83	1.00	0.91
SOUTHERN DOGFACE	0.83	1.00	0.91
VICEROY	0.83	1.00	0.91
YELLOW SWALLOW TAIL	0.83	1.00	0.91
ATALA	0.83	1.00	0.91
ATLAS MOTH	0.83	1.00	0.91
ADONIS	0.83	1.00	0.91
CHECQUERED SKIPPER	0.83	1.00	0.91
COMMON BANDED AWL	0.83	1.00	0.91
ORANGE TIP	0.83	1.00	0.91
ORCHARD SWALLOW	0.83	1.00	0.91
GREEN CELLED CATTLEHEART	0.83	1.00	0.91
JULIA	0.83	1.00	0.91
METALMARK	0.83	1.00	0.91
PAINTED LADY	0.83	1.00	0.91

3.4. Modelo preentrenado ResNet

ResNet, conocida como Redes Residuales, es una arquitectura de red neuronal convolucional lanzada en 2015. Esta arquitectura tiene capacidad de desarrollar y entrenar eficientemente redes neuronales de gran profundidad. Su característica principal radica en las conexiones residuales que implementa, proporcionando atajos que permiten el paso de señales a través de las capas. Esto es clave para conseguir disminuir la degradación del aprendizaje asociado a redes más profundas. ResNet es eficaz en numerosas tareas de visión por computadora, destacando en el manejo y aprendizaje de grandes volúmenes de datos y en la identificación de características visuales complejas.

Tras realizar diversas pruebas con distintas variantes se ha seleccionado la variante ResNet-101V2, que presenta mejoras significativas respecto a las versiones anteriores. Se caracteriza por tener 101 capas, ofreciendo un mayor nivel de profundidad y capacidad de procesamiento, además de optimizar la estructura de capas residuales y de activación, teniendo como resultado un entrenamiento más eficiente. Su diseño profundo permite aprender una gran variedad de características de imágenes de conjuntos de datos grandes, es capaz de capturar y aprender una diversidad más amplia de patrones y características visuales, lo que hace que sea extremadamente eficiente para tareas de clasificación y reconocimiento de imágenes.

3.4.1. Configuraciones analizadas

La estructura del modelo creado para el entrenamiento ResNet tiene la estructura mostrada en la Figura 16.

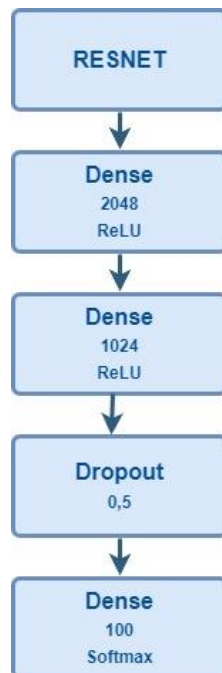


Figura 16: Arquitectura con modelo ResNet

- **Modelo Base:** El primer paso implica la elección de una variante específica de ResNet, en este caso ResNet101V2, cada una tendrá su propia configuración de capas y arquitectura. Mediante la técnica de transferencia de aprendizaje, las capas superiores de clasificación del modelo original no se incluyen, lo que permite una personalización más adecuada para el proyecto en cuestión. Se emplean los pesos entrenados en ImageNet, aprovechando así el aprendizaje previo. Además, se aplica un pooling promedio global después de las capas convolucionales para condensar las características importantes en un formato más compacto y manejable, facilitando el proceso de clasificación.
- **Adición de Capas Densas:** Tras realizar varias pruebas con distintas configuraciones de capas densas, finalmente se añaden dos, una con 2048 y otra de 1024 neuronas, utilizando la función de activación *ReLU* para procesar las características obtenidas del modelo base. Estas capas tienen la tarea de transformar las características en una forma adecuada para la clasificación. Para contrarrestar el sobreajuste, se incluye una capa de Dropout con una tasa del 50%, lo que ayuda a mejorar la capacidad del modelo para generalizar a nuevos datos.
- **Capa de Salida:** La última etapa del modelo es una capa densa compuesta por 100 neuronas, correspondiendo cada una a una clase distinta, y se utiliza una función de activación *softmax*. Esta capa transforma las características procesadas en un conjunto de probabilidades, facilitando la determinación de la clase más probable para cada imagen. La función *softmax* es especialmente adecuada para tareas de clasificación multiclase, permitiendo una clasificación precisa y confiable.

3.4.2. Entrenamiento

Tras realizar diversos experimentos modificando el optimizador como la tasa de entrenamiento el mejor resultado se obtiene con el optimizador SGD partiendo con la tasa de aprendizaje por defecto (0.01).

Del entrenamiento se puede observar:

- Hay mejora en métricas a lo largo de las épocas.
Las métricas indican un aprendizaje efectivo y una buena generalización, y los mecanismos de control como la reducción de la tasa de aprendizaje y la detención temprana ayudan a maximizar la eficiencia del entrenamiento:
 - **Precisión Categórica (Figura 17):** Aumenta significativamente desde un 51.7% en la primera época hasta un 98.73% en la décima época en el conjunto de entrenamiento. Esto indica que el modelo clasifica cada vez mejor las imágenes.

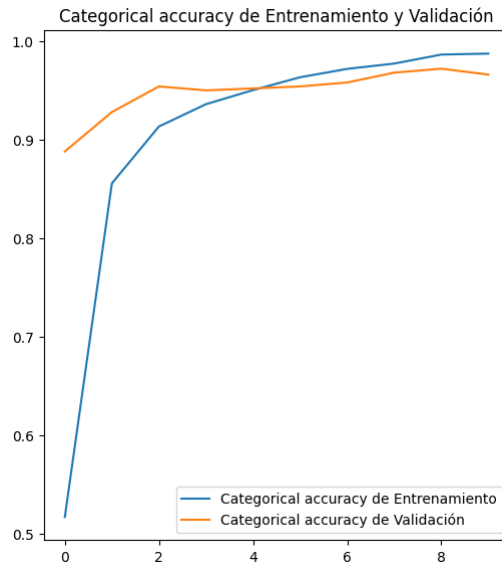


Figura 17: Categorical accuracy modelo ResNet101V2.
Fuente:Cuaderno Colab

- Pérdida (Loss) (Figura 18): Su valor disminuye de manera constante, lo cual es un indicador de que el modelo está aprendiendo de forma efectiva.

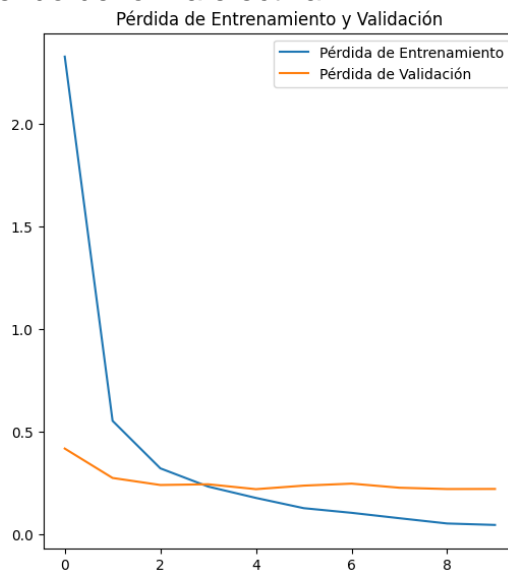


Figura 18: Pérdida modelo ResNet101V2.
Fuente:Cuaderno Colab

- Precision (Figura 19) y Recall (Figura 20): Ambas métricas son altas y mejoran con el tiempo, lo que sugiere que el modelo es bueno tanto en identificar las clases correctas (precisión) como en cubrir una alta proporción de las instancias de cada clase (recall).

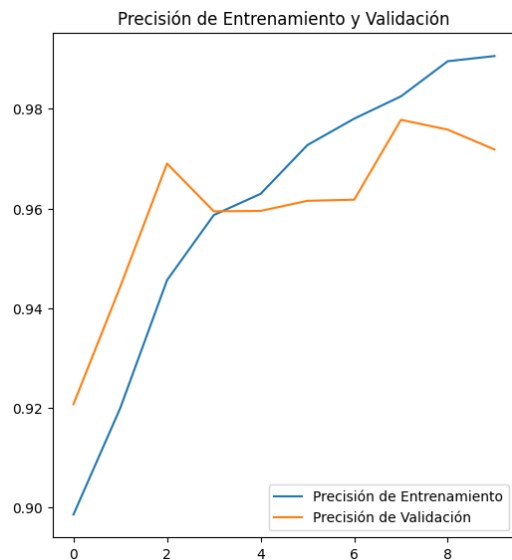


Figura 19: Precisión modelo ResNet101V2.
Fuente: Cuaderno Colab.

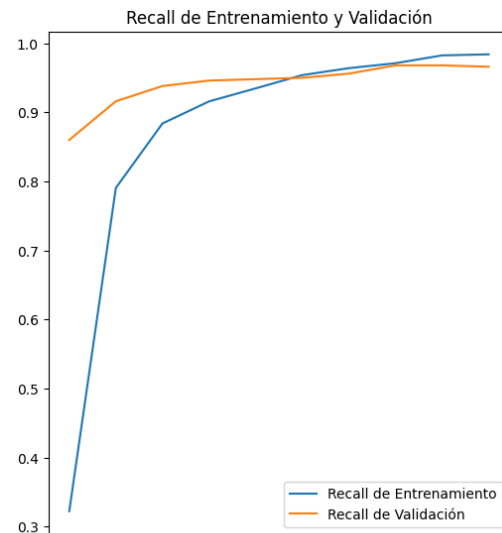


Figura 20: Recall modelo ResNet101V2.
Fuente: Cuaderno Colab

- F1 Score (Figura 21): Aumenta constantemente, lo que refleja un buen equilibrio entre precisión y recall.

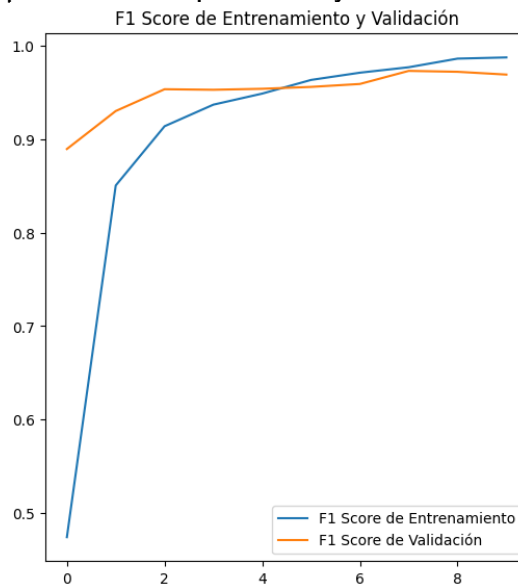


Figura 21: F1-score modelo ResNet101V2. Fuente: Cuaderno Colab

- Reducción de la Tasa de Aprendizaje en la época 8, momento en el que el modelo deja de mejorar con la tasa actual durante tres épocas consecutivas.
- Detención temprana en la época 10, el modelo dejó de mejorar en el conjunto de validación durante 5 épocas consecutivas, por ello se produce la detención para evitar sobreajuste.
- Las métricas en el conjunto de validación son también muy altas, lo que indica que el modelo además de aprender también generaliza bien con nuevos datos.

3.4.3. Evaluación y predicción

Tras evaluar el modelo con el conjunto de datos de test se obtienen los siguientes resultados:

Categorical accuracy	0.9739999771118164
Loss	0.09343817830085754
Precision	0.9798792600631714
Recall	0.9739999771118164
F1 Score	0.9769307374954224

- Precisión Categórica (Categorical Accuracy): Un valor de 97.40% es bastante alto, especialmente para una tarea de clasificación con 100 clases. Esto significa que el modelo es capaz de clasificar correctamente una gran mayoría de las imágenes.
- Pérdida (Loss): Un valor de 0.0934 es bastante bajo, lo que indica que el modelo realiza predicciones precisas.
- Precisión (Precision): Esta métrica indica que de todas las clases que el modelo ha predicho, casi el 97.99% son correctas. Es un indicador de que hay muy pocos falsos positivos.
- Recall: El modelo es capaz de identificar correctamente el 97.40% de todas las instancias positivas reales.
- F1 Score: Con un valor de 0.9769, muestra un excelente equilibrio entre precisión y recall, lo cual es ideal en la mayoría de las situaciones de clasificación.

Tras realizar predicción con el conjunto de datos de test se ha obtenido tanto la matriz de confusión (Figura 22) como el informe de clasificación. Se puede observar que 18 de las 100 especies no han conseguido una predicción perfecta, aunque sí casi perfecta.

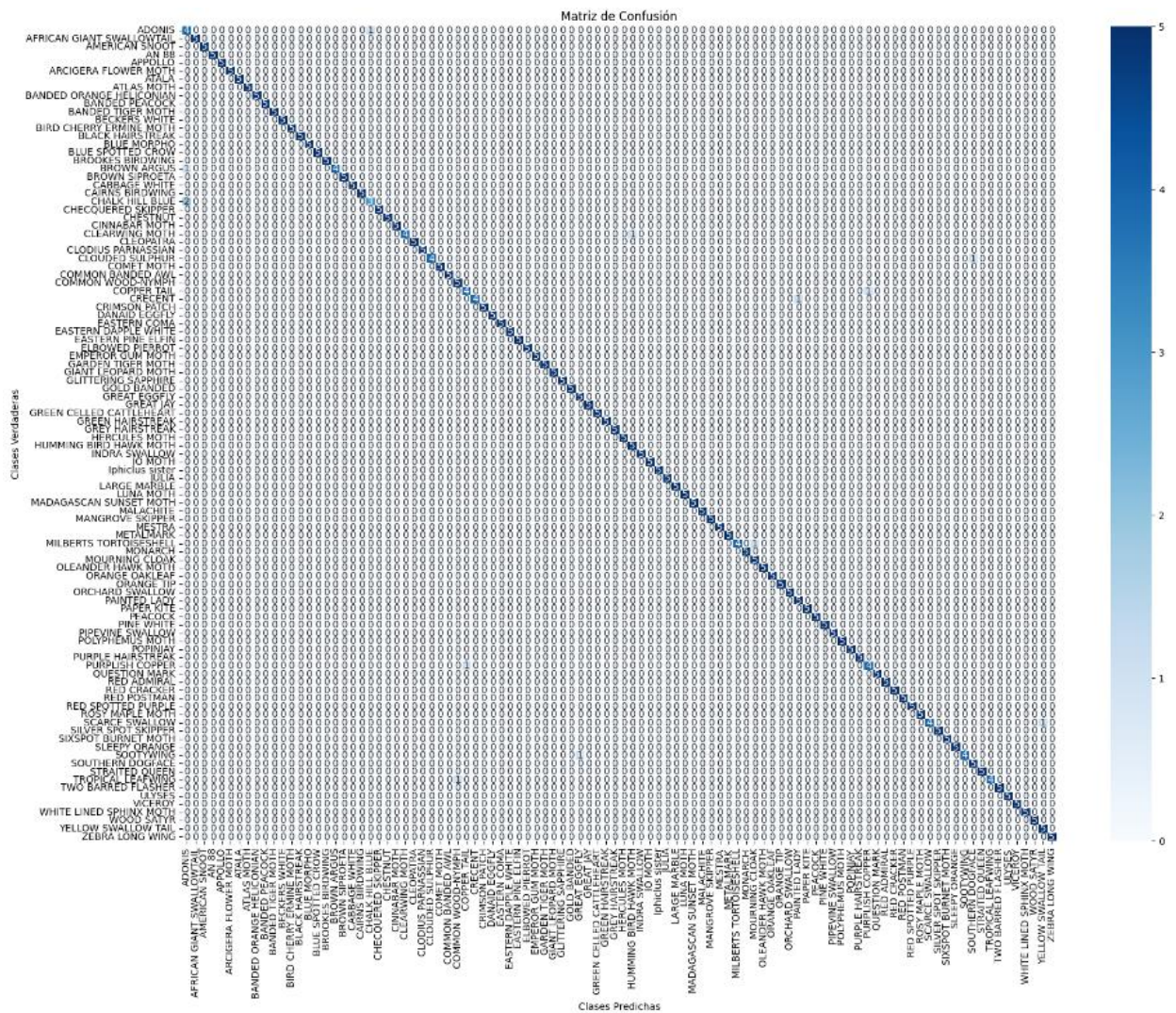


Figura 22: Matriz de confusión ResNet. Fuente: Cuaderno Colab

Especie	Precision	Recall	F1-score
ADONIS	0.57	0.80	0.67
CHALK HILL BLUE	0.75	0.60	0.67
COPPER TAIL	0.80	0.80	0.80
PURPLISH COPPER	0.80	0.80	0.80
COMMON WOOD-NYMPH	0.83	1.00	0.91
GREAT EGGFLY	0.83	1.00	0.91
HUMMING BIRD HAWK MOTH	0.83	1.00	0.91
MOURNING CLOAK	0.83	1.00	0.91
PAINTED LADY	0.83	1.00	0.91
SOUTHERN DOGFACE	0.83	1.00	0.91
YELLOW SWALLOW TAIL	0.83	1.00	0.91
CLEARWING MOTH	1.00	0.80	0.89
CLOUDED SULPHUR	1.00	0.80	0.89
CRECENT	1.00	0.80	0.89
MILBERTS TORTOISESHELL	1.00	0.80	0.89
SCARCE SWALLOW	1.00	0.80	0.89
SOOTYWING	1.00	0.80	0.89
TROPICAL LEAFWING	1.00	0.80	0.89

3.5. Modelo preentrenado EfficientNet

EfficientNet, desarrollada en 2019 por Mingxing Tan y Quoc V de Google Research, es una arquitectura de red neuronal convolucional que destaca en la clasificación de imágenes. Su novedad radica en el "Escalado Compuesto", balanceando profundidad, anchura y resolución para una eficiencia óptima. Con variantes de B0 a B7, emplea bloques MBConv, que combinan convoluciones separables por profundidad con conexiones residuales para reducir la carga computacional. Incorpora también la activación Swish, más efectiva que ReLU en redes profundas, mejorando la activación neuronal. Estas innovaciones hacen a EfficientNet eficiente y precisa en el procesamiento de imágenes complejas.

Tras realizar experimentos con todas las variantes se ha seleccionado EfficientNet B5. Esta variante ha dado mejores resultados que las comprendidas entre la B0 a B4 y no se han usado las posteriores B6 y B7 ya que exigen una alta capacidad computacional de la que no dispone en la plataforma Colab.

3.5.1. Configuraciones analizadas

La estructura del modelo creado para el entrenamiento EfficientNet tiene la estructura mostrada en la Figura 23.



Figura 23: Arquitectura con modelo EfficientNet

- **Modelo Base:** El primer paso implica la elección de una variante específica de EfficientNet, en este caso EfficientNetB5, cada una tendrá su propia configuración de capas y arquitectura. Mediante la técnica de transferencia de aprendizaje, las capas superiores de clasificación del modelo original no se incluyen, lo que permite una personalización más

adecuada para el proyecto en cuestión. Se emplean los pesos entrenados en ImageNet, aprovechando así el aprendizaje previo. Además, se aplica un pooling promedio global después de las capas convolucionales para condensar las características importantes en un formato más compacto y manejable, facilitando el proceso de clasificación.

- **Adición de Capas Densas:** Tras realizar varias pruebas con distintas configuraciones de capas densas, finalmente se añaden dos, una con 4096 y otra de 2048 neuronas, utilizando la función de activación *ReLU* para procesar las características obtenidas del modelo base. Estas capas tienen la tarea de transformar las características en una forma adecuada para la clasificación. Para contrarrestar el sobreajuste, se incluye una capa de Dropout con una tasa del 50%, lo que ayuda a mejorar la capacidad del modelo para generalizar a nuevos datos.
- **Capa de Salida:** La última etapa del modelo es una capa densa compuesta por 100 neuronas, correspondiendo cada una a una clase distinta, y se utiliza una función de activación *softmax*. Esta capa transforma las características procesadas en un conjunto de probabilidades, facilitando la determinación de la clase más probable para cada imagen.

3.5.2. Entrenamiento

Tras realizar diversos experimentos modificando el optimizador como la tasa de entrenamiento el mejor resultado se obtiene con el optimizador SGD partiendo con la tasa de aprendizaje por defecto (0.01).

Del entrenamiento se puede observar:

- Hay mejora en métricas a lo largo de las épocas. Las métricas indican un aprendizaje efectivo y una buena generalización, y los mecanismos de control como la reducción de la tasa de aprendizaje y la detención temprana ayudan a maximizar la eficiencia del entrenamiento:
 - **Precisión Categórica (Figura 24):** Su valor aumenta de 8.84% a 97.04%, mostrando una mejora continua en la clasificación correcta.

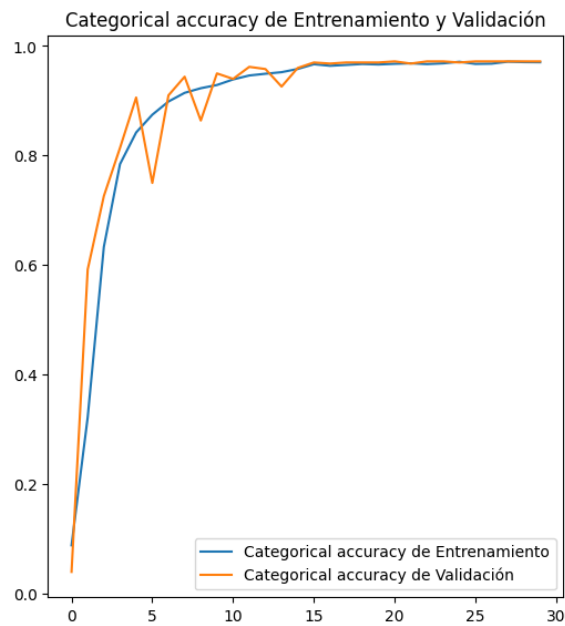


Figura 24: Categorical accuracy modelo EfficientNetB5.
Fuente: Cuaderno Colab

- Pérdida (Loss) (Figura 25): Comienza en 4.5094 y disminuye progresivamente, alcanzando 0.1050, lo que indica un aprendizaje efectivo.

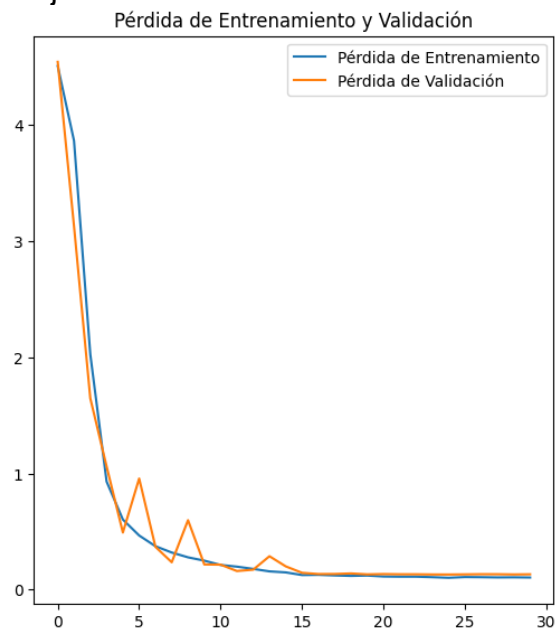


Figura 25: Pérdida modelo EfficientNetB5. Fuente: Cuaderno Colab

- Precision (Figura 26) y Recall (Figura 27): La precisión, comenzó en 97.99% y terminó en 97.87%. El recall, que evalúa la capacidad del modelo para identificar todos los casos correctos, aumentó de 3.72% a 96.22%. Estos cambios indican que el modelo es consistentemente preciso y mejoró significativamente en reconocer correctamente una gran proporción de casos verdaderos.

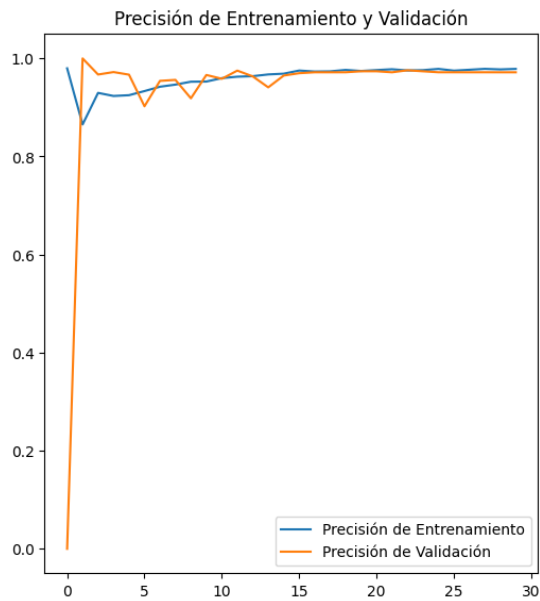


Figura 26: Precision modelo EfficientNetB5.
Fuente: Cuaderno Colab

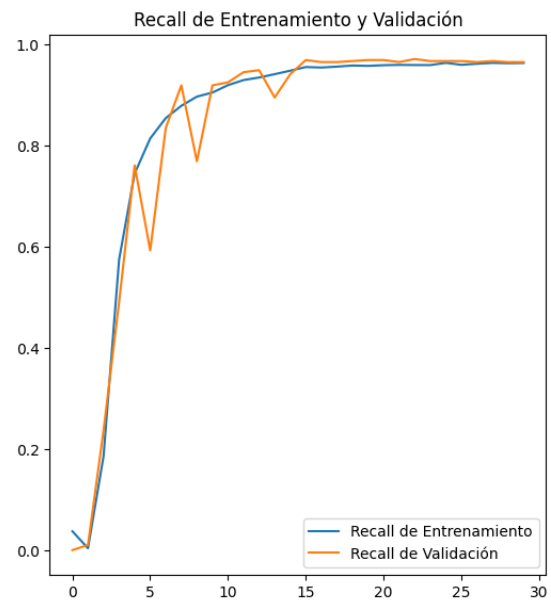


Figura 27: Recall modelo EfficientNetB5.
Fuente: Cuaderno Colab

- F1 Score (Figura 28): Mejora de 0.0717 a 0.9704, reflejando un equilibrio entre precisión y recall.

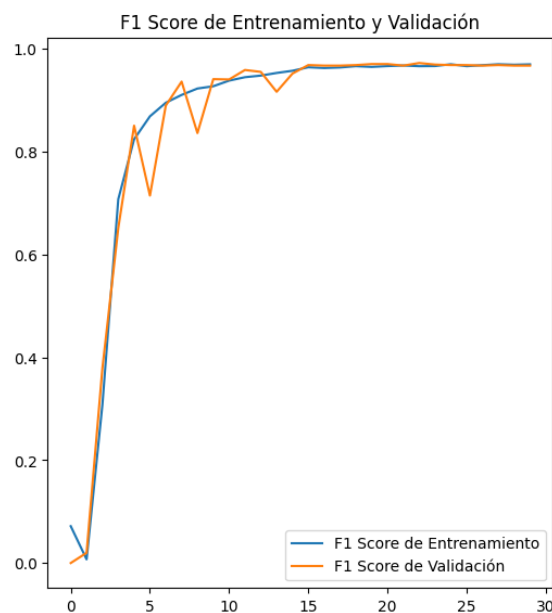


Figura 28: F1-score modelo EfficientNetB5. Fuente: Cuaderno Colab

- Reducción de la Tasa de Aprendizaje en las épocas 15 y 28, momento en el que el modelo deja de mejorar con la tasa actual durante tres épocas consecutivas. Con esta reducción se trata de hacer un ajuste más fino de los pesos para que haya un mejor rendimiento y no se estanque.
- Detención temprana en la época 30, el modelo dejó de mejorar en el conjunto de validación durante 5 épocas consecutivas, por ello se

produce la detención para evitar sobreajuste. Además, ha coincidido esta detención justo al finalizar la última época de las 30 que iba a iterar el entrenamiento.

- Las métricas en el conjunto de validación son también muy altas, lo que indica que el modelo además de aprender también generaliza bien con nuevos datos.

3.5.3. Evaluación y predicción

Categorical accuracy	0.9700000286102295
Loss	0.0916166678071022
Precision	0.97773277759552
Recall	0.9660000205039978
F1 Score	0.9718309640884399

- Pérdida (Loss): El nivel de pérdidas es bajo (0.0916), indicador de que el modelo realiza predicciones bastante reales.
- Precisión Categórica (Categorical Accuracy): Un valor del 97% es muy alto, esto significa que el modelo clasifica correctamente la gran mayoría de las imágenes.
- Precisión (Precision) y Recall: De todas las clases que el modelo ha predicho casi el 98% son correctas. También es capaz de identificar correctamente el 96.60% de todas las instancias positivas reales.
- F1-score: Un F1-score de 0.9718 indica un excelente equilibrio entre la precisión y el recall.

Tras realizar predicción con el conjunto de datos de test se ha obtenido tanto la matriz de confusión (Figura 29): como el informe de clasificación. Se puede observar que 21 de las 100 especies no han conseguido una predicción perfecta.

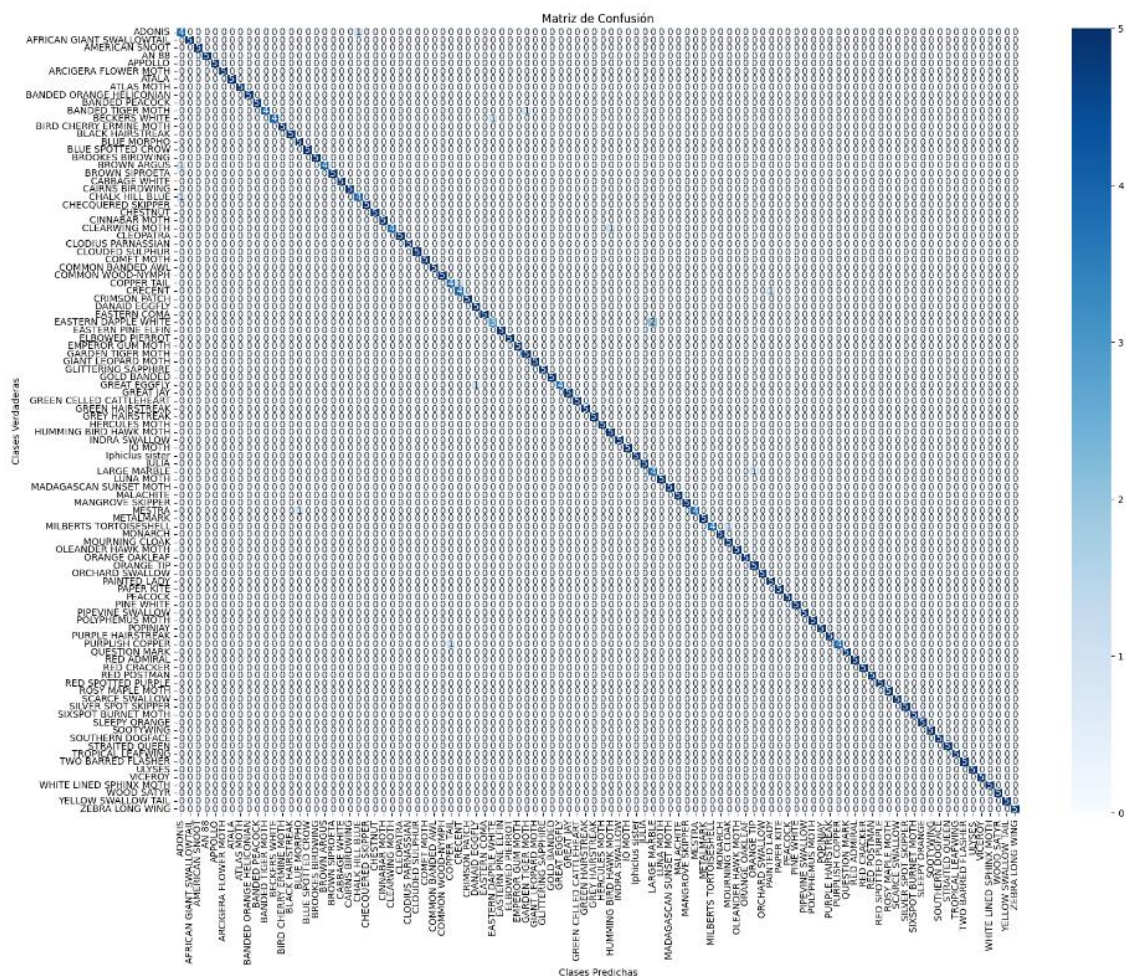


Figura 29: Matriz de confusión EfficientNetB5. Fuente: Cuaderno Colab

Especie	Precision	Recall	F1-score
EASTERN DAPPLE WHITE	0.75	0.60	0.67
LARGE MARBLE	0.67	0.80	0.73
ADONIS	0.67	0.80	0.73
COPPER TAIL	0.80	0.80	0.80
CRECENT	0.80	0.80	0.80
CHALK HILL BLUE	0.80	0.80	0.80
BANDED TIGER MOTH	1.00	0.80	0.89
BECKERS WHITE	1.00	0.80	0.89
CLEARWING MOTH	1.00	0.80	0.89
MESTRA	1.00	0.80	0.89
MILBERTS TORTOISESHELL	1.00	0.80	0.89
PURPLISH COPPER	1.00	0.80	0.89
BROWN ARGUS	1.00	0.80	0.89
PURPLISH COPPER	1.00	0.80	0.89
BLUE MORPHO	0.83	1.00	0.91
GARDEN TIGER MOTH	0.83	1.00	0.91
HUMMING BIRD HAWK MOTH	0.83	1.00	0.91
MOURNING CLOAK	0.83	1.00	0.91

ORANGE TIP	0.83	1.00	0.91
PAINTED LADY	0.83	1.00	0.91
DANAID EGGFLY	0.83	1.00	0.91

4. Resultados

- Entrenamiento y Validación
 - Modelo Secuencial: Mejora constante, pero la precisión final y las otras métricas son inferiores a los otros dos modelos.
 - ResNet101V2: Rápida convergencia y alto rendimiento en el conjunto de validación, con una precisión y métricas generales muy altas.
 - EfficientNetB5: Comenzó con un rendimiento más bajo, pero mejoró significativamente a lo largo del entrenamiento, alcanzando un rendimiento similar a ResNet101V2.
- Evaluación
 - Modelo Secuencial: Este modelo muestra la mayor pérdida y la menor precisión categórica y F1-score en comparación con los otros dos modelos. Este modelo tiene un margen amplio de mejora en términos de precisión general y exactitud en las predicciones.
 - ResNet101V2: Tiene el mejor equilibrio entre las métricas evaluadas. Tiene una pérdida similar a EfficientNetB5, pero ligeramente superior, en cambio tiene la precisión categórica más alta y el F1-score más alto (97.6%), este modelo demuestra un rendimiento general superior en términos de precisión y fiabilidad de sus predicciones, haciéndolo ideal para situaciones donde la precisión es imprescindible.
 - EfficientNetB5: Tiene la pérdida más baja, pero su precisión categórica y F1-score son ligeramente inferiores a los de ResNet101V2. Este modelo es muy eficiente en minimizar errores en sus predicciones, aunque con un ligero compromiso en la precisión y el balance entre precisión y recall en comparación con ResNet101V2.

	Secuencial	ResNet101V2	EfficientNetB5
Loss	0.3558	0.0934	0.0916
Categorical Accuracy	90.80%	97.40%	97.00%
F1-Score	91,3%	97,6%	97,1%

- Predicción
 - Modelo Secuencial: Precisión del 91%, con algunas clases que tienen una baja puntuación de precisión y recall.
 - ResNet101V2: Precisión del 97%, con una alta consistencia en la precisión y el recall para casi todas las clases.
 - EfficientNetB5: Precisión del 97%, con rendimiento similar a ResNet101V2 en la mayoría de las clases.

	Secuencial	ResNet101V2	EfficientNetB5
Accuracy	91%	97%	97%
Promedio Precision	92%	98%	97%
Promedio Recall	91%	97%	97%
Promedio F1-score	91%	97%	97%
Número de especies con predicciones imperfectas	58	18	21

5. Conclusiones y trabajos futuros

5.1. Conclusiones

Tras realizar el trabajo se ha comprobado que las redes neuronales son capaces de clasificar imágenes con una alta eficacia y precisión, como se demuestra en la comparación entre los modelos Secuencial, ResNet101V2 y EfficientNetB5 para clasificación multiclase. ResNet101V2 y EfficientNetB5 destacan por su superioridad, superando al modelo secuencial en la mayoría de las métricas. Estos modelos también muestran una mejor consistencia y capacidad de generalización, elementos esenciales para la clasificación efectiva de múltiples categorías.

Los modelos preentrenados, como ResNet101V2 y EfficientNetB5, ofrecen claros beneficios en términos de eficiencia y adaptabilidad, gracias al aprendizaje transferido y su habilidad para adaptarse a nuevos datos. Por otro lado, el modelo secuencial, aunque pueda ser valioso para fines educativos, requiere un conocimiento más profundo en modelado y ajuste fino, lo que puede ser un desafío sin experiencia significativa en el tema.

En resumen, los modelos ResNet101V2 y EfficientNetB5 se destacan como opciones superiores en la clasificación multiclase, proporcionando alta precisión y eficiencia. Estos modelos se posicionan como soluciones robustas y accesibles para tareas avanzadas de clasificación. Además, el empleo de modelos preentrenados como estos facilita enormemente la clasificación de lepidópteros o cualquier otro insecto, un aspecto vital para la investigación biológica y los esfuerzos de conservación. Esta mejora en la eficiencia permite una identificación más rápida y precisa de las especies.

En cuanto a la planificación ha sido necesario realizar modificaciones, tanto por motivos externos ya indicados en los riesgos, como por inexperiencia en cuando

En el transcurso del proyecto, se han producido desviaciones de la planificación inicial. Estas desviaciones pueden atribuirse en parte a factores externos imprevistos que han impactado al progreso del proyecto. Además, es importante reconocer que la inexperiencia en el desarrollo y entrenamiento de redes neuronales también ha jugado un papel significativo. La naturaleza impredecible de trabajar con estas tecnologías avanzadas en combinación con la curva de aprendizaje en este campo ha requerido que se realicen ajustes continuos en la planificación para la adaptación mejor a las necesidades y a los hallazgos inesperados.

5.2. Trabajos futuros

Existe varias posibles líneas futuras relacionadas a este proyecto:

- Detección y Segmentación de Imágenes además de la clasificación, se puede trabajar en la detección y segmentación de las mariposas y polillas en imágenes complejas, lo que permitiría identificar y clasificar múltiples especímenes en una sola imagen.
- Aplicación Móvil para la Identificación en Campo: Desarrollar una aplicación móvil que utilice el modelo de clasificación para permitir a los usuarios identificar mariposas y polillas en tiempo real.
- Integración con bases de datos científicas para contribuir en la investigación y conservación de estas especies.
- Explorar métodos de aprendizaje no supervisado o supervisado para hacer uso de imágenes no etiquetadas.

6. Glosario

- **Accuracy** (Exactitud): Mide el porcentaje de predicciones correctas del modelo.
- **Categorical Accuracy** (Exactitud Categórica): Exactitud en problemas de clasificación multiclase.
- **Clasificación**: Asignar categorías a datos basados en características.
- **CNN** (Convolutional Neural Network, Red Neuronal Convolutacional): Usada en procesamiento de imágenes.
- **Colab**: Plataforma de Google que proporciona un entorno de notebook basado en la nube para Python.
- **Convolución**: Operación matemática fundamental en las CNN que procesa datos con filtros.
- **Data Augmentation** (Aumento de Datos): Técnica para crear datos adicionales mediante modificaciones de los datos existentes.
- **Dataset** (Conjunto de Datos): Utilizado para entrenar y evaluar modelos.
- **EfficientNet**: Arquitectura de red neuronal diseñada para optimizar la eficiencia computacional.
- **EfficientNetB5**: Versión B5 de la arquitectura EfficientNet.
- **Epoch** (Época): Un pase completo a través del conjunto de entrenamiento.
- **Gantt**: Herramienta de gestión de proyectos que muestra el cronograma de las actividades.
- **Gdrive** (Google Drive): Servicio de almacenamiento en la nube de Google.
- **Google**: Empresa especializada en servicios relacionados con Internet.
- **GPU** (Graphics Processing Unit, Unidad de Procesamiento Gráfico): Eficiente para el cálculo en deep learning.
- **Input** (Entrada): Datos que se alimentan a un modelo.
- **JPEG**: Formato de compresión para imágenes.
- **Keras**: Biblioteca de Python para redes neuronales.
- **Learning Rate** (Tasa de Aprendizaje): Define el tamaño del paso en el descenso del gradiente.
- **Loss Function** (Función de Pérdida): Mide el error entre las predicciones y los valores reales.
- **Matriz de Confusión**: Muestra las predicciones correctas e incorrectas de un modelo.
- **Max Pooling** (Agrupamiento Máximo): Técnica para reducir dimensionalidad en redes neuronales.
- **Neural Network** (Red Neuronal): Modelo computacional diseñado para procesar información.
neuronal.
- **Output** (Salida): Los resultados producidos por un modelo.
- **Overfitting** (Sobreajuste): El modelo se ajusta demasiado a los datos de entrenamiento.
- **Precision** (Precisión): Proporción de predicciones positivas correctas.

- **Python:** Lenguaje de programación utilizado en ciencia de datos y aprendizaje automático.
- **Recall** (Recuperación): Proporción de casos positivos reales identificados correctamente.
- **ReLU** (Rectified Linear Unit, Unidad Lineal Rectificada): Función de activación en redes neuronales.
- **ResNet** (Residual Network, Red Neuronal Residual): Red neuronal con conexiones de salto.
- **ROC** (Receiver Operating Characteristic, Característica Operativa del Receptor): Curva que muestra la capacidad de un modelo de clasificación.
- **Softmax:** Función de activación que convierte salidas en probabilidades.
- **Test Dataset** (Conjunto de Datos de Prueba): Utilizado para probar el rendimiento de un modelo.
- **TFM** (Trabajo de Fin de Máster)
- **Training Dataset** (Conjunto de Datos de Entrenamiento): Utilizado para entrenar un modelo.
- **Transfer Learning** (Aprendizaje por Transferencia): Uso de un modelo preentrenado en un problema similar.
- **Validation Dataset** (Conjunto de Datos de Validación): Utilizado para validar el rendimiento de un modelo.

7. Bibliografía

- [1] Culin, J. (2023, September 12). lepidopteran. Encyclopedia Britannica. <https://www.britannica.com/animal/lepidopteran> (Consultado: el 7 de octubre de 2023)
- [2] Rumana Yasmin, Aparna Das, Liton Jude Rozario, Md. Ezharul Islam, Butterfly detection and classification techniques: A review, Intelligent Systems with Applications, Volume 18, 2023, 200214, ISSN 2667-3053, <https://www.sciencedirect.com/science/article/pii/S266730532300039X> (Consultado: el 7 de octubre de 2023)
- [3] Rajeeva P. P., F.; Orban, R.; Vadivel, K.S.; Subramanian, M.; Muthusamy, S.; Elminaam, D.S.A.; Nabil, A.; Abulaigh, L.; Ahmadi, M.; Ali, M.A.S. A Novel Method for the Classification of Butterfly Species Using Pre-Trained CNN Models. Electronics 2022, 11, 2016. <https://doi.org/10.3390/electronics11132016> (Consultado: el 8 de octubre de 2023)
- [4] Toke T. Høye, Johanna Ärje, Kim Bjerger, Oskar L. P. Hansen, Alexandros Iosifidis, Florian Leese, Hjalte M. R. Mann, Kristian Meissner, Claus Melvad, and Jenni Raitoharju. Edited by Matthew L. Forister, University of Nevada, Reno, NV, and accepted by Editorial Board Member May R. Berenbaum October 23, 2020 (received for review March 24, 2020) [Deep learning and computer vision will transform entomology](#) (Consultado: el 8 de octubre de 2023)
- [5] Butterfly Conservation | Xerces Society. (n.d.). Retrieved September 29, 2021 <https://www.xerces.org/endangered-species/butterflies>. (Consultado: el 8 de octubre de 2023)
- [6] D.E. Walter, S. Winterton. Keys and the crisis in taxonomy: Extinction or reinvention? Annual Review of Entomology, 52 (May) (2007), pp. 193-208 <https://www.annualreviews.org/doi/full/10.1146/annurev.ento.51.110104.151054> (Consultado: el 7 de octubre de 2023)
- [7] Kaggle Dataset Butterfly & Moths Image Classification 100 species <https://www.kaggle.com/datasets/gpiosenka/butterfly-images40-species/data>
- [8] Xi, T., Wang, J., Han, Y., Lin, C., and Ji, L. (2022) Multiple butterfly recognition based on deep residual learning and image analysis. *Entomological Research*, 52: 44–53. <https://doi.org/10.1111/1748-5967.12564>.

- [9] Ayad Saad Almryad, Hakan Kutucu, Automatic identification for field butterflies by convolutional neural networks, Engineering Science and Technology, an International Journal, Volume 23, Issue 1, 2020, Pages 189-195, <https://doi.org/10.1016/j.jestch.2020.01.006>.
- [10] Bladon A.J., Bladon E.K., Smith R.K. & Sutherland W.J. (2023) Butterfly and Moth Conservation: Global Evidence for the Effects of Interventions for butterflies and moths. Conservation Evidence Series Synopsis. University of Cambridge, Cambridge, UK
<https://www.conservationevidence.com/synopsis/pdf/37>
- [11] Udemy. Machine Learning y Data Science: Curso completo con Python [última actualización 2023]. Santiago Hernández
- [12] Udemy. Máster Redes Neuronales-Deep Learning con TensorFlow [última actualización 2023]. Iván Pinar Domínguez.
- [13] Udemy. Machine Learning de la A a la Z: Python para data Science [última actualización 2023]. SuperDataScience Team
- [14] Python Deep Learning. Introducción práctica con Keras y TensorFlow 2. [2020] Jordi Torres
- [15] Bjerger K, Nielsen JB, Sepstrup MV, Helsing-Nielsen F, Høye TT. An Automated Light Trap to Monitor Moths (Lepidoptera) Using Computer Vision-Based Tracking and Deep Learning. *Sensors*. 2021; 21(2):343. <https://doi.org/10.3390/s21020343>
- [16] Guías librería Keras
<https://keras.io/guides/>

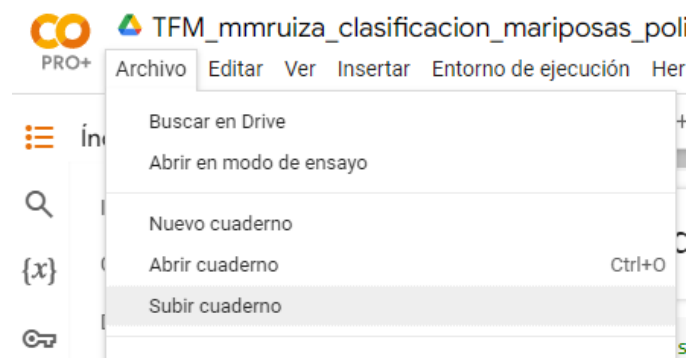
8. Anexos

Se puede acceder al cuaderno Colab con el código fuente a través del siguiente enlace:

https://github.com/mmruiza/TFM/blob/main/TFM_mmruiza_clasificacion_mariposas_polillas.ipynb

El cuaderno contiene la ejecución de las tres mejores configuraciones de los tres modelos entrenados (modelo secuencial, ResNet101V2 y EfficientNetB5).

Para subir el cuaderno a Colab mediante la opción de menú *Archivo -> Subir cuaderno*



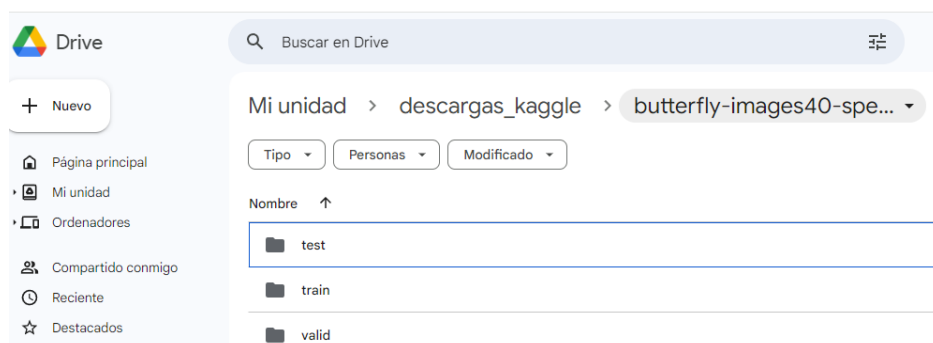
En caso de querer ejecutarlo se deben seguir los primeros pasos:

1. Descargar y descomprimir el dataset comprimido de la web de Kaggle:

<https://www.kaggle.com/datasets/gpiosenska/butterfly-images40-species?resource=download>

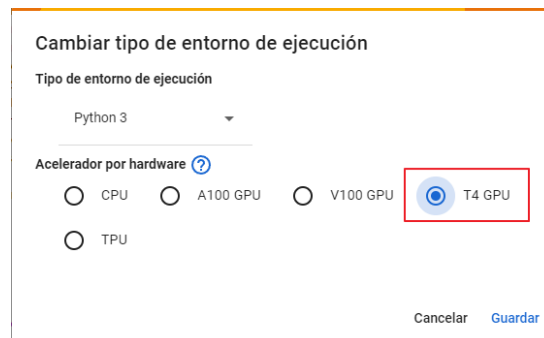
2. Una vez descomprimido se debe situar las carpetas test, valid y train en el directorio de Google Drive:

`/content/gdrive/MyDrive/descargas_kaggle/butterfly-images40-species/`



3. Cambiar el entorno de a T4 GPU mediante la opción:

Entorno de ejecución -> Cambiar tipo de entorno de ejecución del menú



4. Ejecutar el cuaderno pulsando en la opción:

Entorno de ejecución -> Ejecutar todas.

5. Dar permiso a Google Drive para permitir la carga de imágenes cuando se muestre la siguiente ventana:

