

Evaluación de la Exactitud y Precisión por Clase de un Modelo de Red Neuronal Convolutiva en una Aplicación para la Identificación y Búsqueda de Mascotas

Evaluation of Accuracy and Class-wise Precision of a Convolutional Neural Network Model Trained with Teachable Machine in a Pet Identification and Search Application

Mathew Patricio Aviles Chavez ¹[0009-0001-2169-924X], Jaime David Camacho Castillo ²[0000-0002-9110-6585],
Jonny Israel Guaiña Yungán ³[0000-0003-0456-7429], Edgar Edison Barragán Del Pozo ⁴[0009-0002-8208-2726]

¹⁻⁴ Escuela Superior Politécnica de Chimborazo (ESPOCH), Ecuador

¹ mathew.aviles@epoch.edu.ec, ² jaimed.camacho@epoch.edu.ec, ³ jonny.guaina@epoch.edu.ec,
⁴ e_barragan@epoch.edu.ec

CITA EN APA:

Aviles Chavez, M. P., Camacho Castillo, J. D., Guaiña Yungán, J. I., & Barragán Del Pozo, E. E. (2025). Evaluación de la Exactitud y Precisión por Clase de un Modelo de Red Neuronal Convolutiva en una Aplicación para la Identificación y Búsqueda de Mascotas. *Tesla Revista Científica*, 5(1), e428. <https://doi.org/10.55204/trc.v5i1.e428>

Recibido: 2025-02-25

Revisado: 2025-03-01 al 2025-03-27

Corregido: 2025-04-10

Aceptado: 2025-04-14

Publicado: 2025-05-21

TESLA

Revista Científica

ISSN: 2796-9320



Los contenidos de este artículo están bajo una licencia de Creative Commons Attribution 4.0 International (CC BY 4.0)

Los autores conservan los derechos morales y patrimoniales de sus obras. The contents of this article are under a Creative Commons Attribution 4.0 International (CC BY 4.0) license. The authors retain the moral and patrimonial rights of their works.

Resumen: En este estudio, se aborda la problemática de la búsqueda de mascotas extraviadas mediante el uso de redes neuronales convolucionales, específicamente implementadas en la plataforma Teachable Machine de Google. Se destaca la limitación de los métodos tradicionales, como redes sociales y carteles físicos, y se propone mejorar la identificación y búsqueda de mascotas perdidas a través de la inteligencia artificial. La metodología implica la recopilación de un conjunto de datos compuesto por imágenes de gatos y perros, utilizando la plataforma Teachable Machine para entrenar el modelo de red neuronal convolutiva. Se realiza una evaluación de la precisión del modelo con conjuntos de datos de diferentes tamaños (10, 30 y 50 imágenes). Los resultados muestran que la precisión del modelo varía significativamente entre los conjuntos de datos, destacando la importancia de proporcionar un mayor número de imágenes para mejorar la precisión. Se sugiere asegurar condiciones óptimas de luminosidad en las imágenes, resaltar rasgos distintivos de las mascotas y mantener consistencia en la cantidad de imágenes en los conjuntos de datos. También se destaca la importancia de considerar la dimensión de reescalado de las imágenes originales durante el desarrollo del modelo.

Palabras clave: software, Python, mascotas, aplicación web, inteligencia artificial, búsqueda, redes neuronales convolucionales.

Abstract: In this study, the issue of searching for lost pets is addressed through the use of convolutional neural networks, specifically implemented on Google's Teachable Machine platform. The limitation of traditional methods, such as social networks and physical posters, is highlighted, and there is a proposal to enhance the identification and search for lost pets through artificial intelligence. The methodology involves gathering a dataset composed of images of cats and dogs, using the Teachable Machine platform to train the convolutional neural network model. An evaluation of the model's accuracy is conducted with datasets of different sizes (10, 30, and 50 images). The results show that the model's accuracy varies significantly among the datasets, emphasizing the importance of providing a greater number of images to improve accuracy. It is suggested to ensure optimal lighting conditions in the images, highlight distinctive features of the pets, and maintain consistency in the number of images in the datasets. The importance of considering the dimension of rescaling the original images during the model's development is also highlighted.

Keywords: software, Python, pets, web application, artificial intelligence, search, convolutional Neural networks.

1. INTRODUCCIÓN

La detección de objetos es una tarea esencial en el procesamiento de imágenes por computadora. Existen muchas aplicaciones en diversos campos como el diagnóstico médico, video vigilancia, seguridad,

etc. La detección de objetos se utiliza como parte integral como la segmentación semántica, la segmentación de instancias, la estimación de poses, la detección de actividades sospechosas, etc (Arulprakash & Aruldoss, 2022).

Los algoritmos de detección de objetos suelen funcionar dividiendo una imagen en una cuadrícula de celdas y utilizando técnicas de aprendizaje automático para analizar las características de los objetos dentro de cada celda. A continuación, el algoritmo predice la presencia y la ubicación de los objetos en cada celda, y estas predicciones se combinan para generar un conjunto final de detecciones de objetos para toda la imagen (Saqib et al., 2025).

El aprendizaje profundo, basado en redes neuronales, es un tipo más avanzado de aprendizaje automático que ofrece soluciones en muchos modelos de aplicaciones complejas utilizando métodos estadísticos tradicionales (Bodero et al., 2020).

A pesar de la disponibilidad de diversos modelos, identificar el modelo más efectivo para casos de uso específicos sigue siendo un desafío, especialmente al tratar con tareas de detección de objetos de múltiples clases. Un desafío clave radica en equilibrar la eficiencia computacional con una alta precisión, particularmente en entornos con recursos limitados, como dispositivos móviles y robótica (Sharma et al., 2022).

En las zonas urbanas de todo el mundo, la confluencia del aumento de la densidad de población y el creciente volumen de objetos perdidos plantea un problema acuciante, agravado por la ineficiencia de los servicios tradicionales de búsqueda manual. Para hacer frente a esta exigencia, se pone de manifiesto la necesidad imperiosa de un rápido desarrollo de sistemas inteligentes de objetos perdidos (Zhou et al., 2023).

Las poblaciones de perros en situación de calle que deambulan libremente por calles y campos son una preocupación para la salud humana. Además, transmiten enfermedades infecciosas algunas de las cuales son zoonóticas y en otros casos causan graves daños a la fauna silvestre (Trujillo Santillán et al., 2024).

Las personas actualmente utilizan ciertas estrategias para buscar a sus mascotas cuando se pierden, la principal estrategia es buscarla en un área cercana, ya que la mayoría de los perros y gatos suelen estar escondidos, por lo que es muy importante revisar áreas cercanas, otra estrategia es el uso de redes sociales y plataformas en línea, ya que publicar en redes sociales, grupos locales y sitios especializados, pueden ayudarnos a aumentar las posibilidades de encontrar a la mascota. Otra estrategia utilizada habitualmente por las personas es el uso de carteles físicos colocados en sitios especializados donde exista una gran afluencia de personas como parques, paradas de bus, tiendas de mascotas y clínicas especializadas pueden ayudar a mejorar las probabilidades de encontrar a la mascota (Andreae & Andreae, 1978). Para mejorar la búsqueda e identificación de mascotas perdidas, se propone el uso de redes neuronales convolucionales. Las Redes Neuronales Convolucionales (CNN) es un tipo de modelo de aprendizaje profundo ampliamente utilizado en el campo de la inteligencia artificial, especialmente para el procesamiento de imágenes. Su principal ventaja es que identifica automáticamente características relevantes sin necesidad de supervisión

humana, lo que la hace fundamental para tareas como visión por computadora, procesamiento del habla y reconocimiento facial (Alzubaidi et al., 2021).

Estos "cerebros" tienen capas que les permiten entender cosas simples al principio, como bordes y colores, y luego capas más profundas que entienden cosas más complejas, como formas y patrones. Las CNN son geniales porque comparten información entre partes cercanas de una imagen y utilizan muchas capas para entender detalles más y más complejos (Massiris et al., 2018).

A diferencia de las redes neuronales artificiales tradicionales (ANN), las CNN optimizan automáticamente la extracción de características relevantes sin intervención humana, lo que las hace altamente eficaces en tareas como visión por computadora, reconocimiento facial y clasificación de objetos. Inspiradas en la corteza visual de los mamíferos, estas redes utilizan capas de convolución, que aplican filtros sobre la imagen para identificar patrones, y capas de agrupamiento (pooling), que reducen la dimensionalidad para mejorar la eficiencia computacional. Su arquitectura, compuesta por múltiples capas de procesamiento, permite modelar relaciones espaciales en los datos de entrada, minimizando la cantidad de parámetros y acelerando el entrenamiento. Este diseño no solo mejora la precisión en el reconocimiento de imágenes, sino que también facilita la implementación en dispositivos con recursos limitados, haciendo de las CNN la tecnología dominante en aplicaciones de inteligencia artificial enfocadas en procesamiento visual (O'shea & Nash, 2015).

En este estudio, se aborda el desafío de desarrollar un modelo de inteligencia artificial para la identificación y búsqueda de mascotas. Para lograr esto, se utilizó la plataforma Teachable Machine, una herramienta diseñada por Google que permite a los usuarios crear modelos de aprendizaje automático de manera accesible y sin la necesidad de una programación complicada.

Las herramientas de AutoML normalmente integran en un pipeline unificado una variedad de etapas de análisis, como limpieza de datos, ingeniería de características, generación de modelos y ajuste de hiperparámetros, ensamblaje de conjuntos y evaluación. Muchos sistemas de AutoML ofrecen enfoques de bajo código o sin código que eliminan la necesidad de experiencia en aprendizaje automático por parte del usuario. Existen opciones de AutoML de empresas establecidas, como Azure ML de Microsoft, CreateML de Apple, AutoML y Teachable Machine de Google (Tjaden & Tjaden, 2023).

Teachable Machine es una herramienta de Google diseñada para entrenar modelos de aprendizaje automático de manera sencilla, sin necesidad de conocimientos avanzados en programación. Su interfaz intuitiva permite a cualquier usuario arrastrar y soltar datos para crear modelos personalizados, los cuales pueden ser utilizados en diversas aplicaciones como reconocimiento de imágenes, audio o poses corporales. El sistema automatiza la creación de modelos de redes neuronales, basándose en la idea de entrenamiento con datos específicos, lo que implica que el rendimiento del modelo dependerá de la calidad y diversidad del conjunto de entrenamiento. Aunque los modelos generados pueden contener cierto sesgo, sirven como una introducción pedagógica a los conceptos fundamentales de la inteligencia artificial y demuestran las limitaciones del aprendizaje automático (Siddiqui, 2023).

Teachable Machine no solo facilita el entrenamiento de modelos de aprendizaje automático sin conocimientos de programación, sino que también permite la personalización avanzada mediante el ajuste de hiperparámetros para mejorar el rendimiento del modelo. Además de ofrecer pre-conjuntos de datos de Google, los usuarios pueden cargar sus propios datos para entrenar modelos más especializados. La herramienta admite diversos formatos de entrada, como imágenes, audio y texto, lo que la hace versátil para aplicaciones en reconocimiento de objetos, clasificación de imágenes y procesamiento de lenguaje natural (Liu et al., 2023).

2. METODOLOGÍA

2.1. Descripción del conjunto de datos utilizado

Los datos que se recopilieron para la generación del dataset corresponden a imágenes de mascotas correspondientes a gatos y perros, las cuales fueron recopiladas en gran parte en la clínica veterinaria VitaFarm de la ciudad de Riobamba, otras fueron tomadas de forma propia, es decir de mascotas pertenecientes a familiares y amigos que aportaron con imágenes de mascotas. El número de muestras es variable, pero se toma como base 50 imágenes para cada clase.

2.2. Teachable Machine

Para el desarrollo de la red neuronal convolucional, aprovechamos la plataforma de Teachable Machine. Teachable Machine es una herramienta de aprendizaje automático desarrollada por Google que permite a los usuarios crear modelos de inteligencia artificial de manera intuitiva y accesible, sin necesidad de experiencia en programación o aprendizaje automático (Kacorri, 2017).

Teachable Machine se distingue por su interfaz amigable, que facilita la creación de modelos personalizados para la clasificación de imágenes. Esta interfaz permite a los usuarios cargar y etiquetar datos de entrenamiento de manera sencilla, simplificando el proceso de enseñar al modelo a reconocer patrones específicos (Prasad & Manjunath, 2024). La accesibilidad de esta herramienta es notable, ya que no requiere conocimientos técnicos avanzados en aprendizaje automático o programación. En consecuencia, Teachable Machine se presenta como una solución inclusiva, permitiendo que personas con diversas habilidades participen efectivamente en la creación de modelos personalizados para sus necesidades de clasificación de imágenes. La plataforma utiliza una arquitectura de red neuronal convolucional subyacente, que se entrena en función de los datos proporcionados. Una vez entrenado, el modelo puede ser exportado y utilizado en diversas aplicaciones, como la identificación y búsqueda de mascotas en nuestro caso. Esta plataforma es particularmente útil para aplicaciones de clasificación de imágenes y se ha utilizado con éxito en una variedad de proyectos de aprendizaje automático de nivel principiante y profesional. Su facilidad de uso y su capacidad para generar modelos personalizados gracias a su interfaz gráfica intuitiva y con varios ajustes preestablecidos que se pueden configurar de manera fácil la convierten en una opción valiosa para nuestro enfoque en la identificación y búsqueda de mascotas (Carney et al., 2020).

2.3. Generación del dataset

Para crear el conjunto de datos se va a determinar el número de imágenes que serán adecuadas para que el modelo pueda tener una precisión considerablemente alta. Para esto se realizó pruebas con tres grupos de datasets con diferentes números de datos, en este caso los grupos van a estar divididos en 10, 30 y 50 imágenes para lograr determinar si existe una diferencia significativa en la precisión al momento de realizar la comparación de imágenes de las mascotas. Para lo cual se realizaron cinco pruebas por cada conjunto de datos, en los cuales se va a registrar el porcentaje de similitud que nos arroje estas pruebas.

2.4. Procesamiento y análisis de la información

Para entrenar el modelo se debe tener listas las imágenes de las mascotas que se va a utilizar en el modelo, para ello se crearán clases o etiquetas las mismas que identifiquen a cada mascota por un nombre para distinguirlas entre sí, tal como se ve en la ilustración 1.

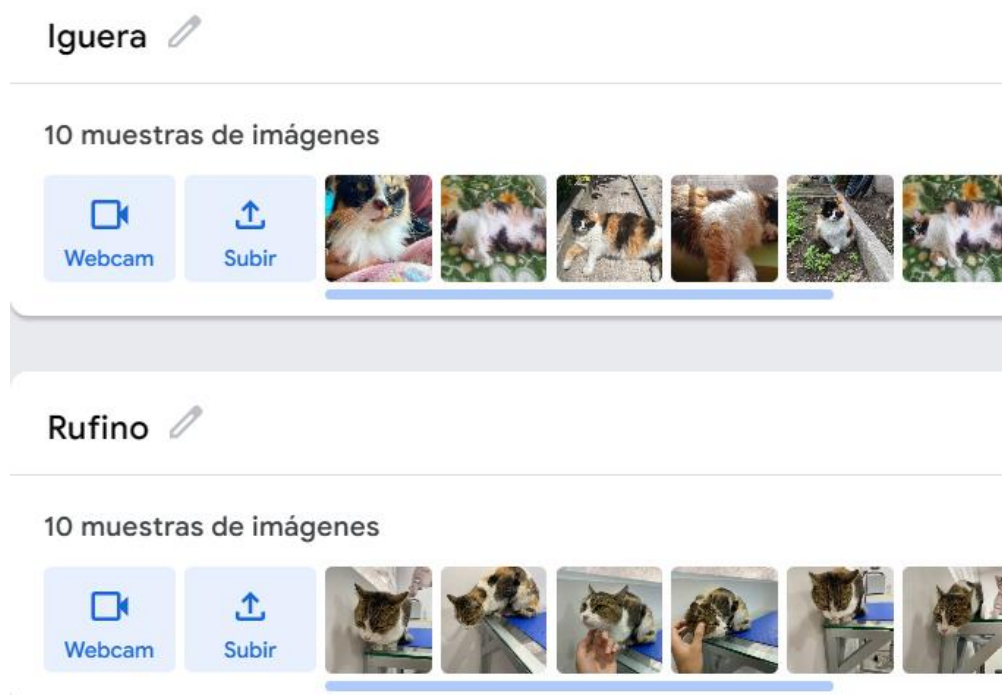


Ilustración 1 Clases de las mascotas

Los datos se encuentran divididos en conjuntos de entrenamiento, validación y prueba. La preparación de la red neuronal está configurada de la siguiente manera: Épocas, Tamaño del lote y tasa de aprendizaje.

- **Épocas:** se refieren al número de veces que el modelo recorre todo el conjunto de entrenamiento durante el proceso de entrenamiento. Cada época representa un ciclo completo a través de todos los datos de entrenamiento, y durante cada ciclo, el modelo ajusta sus pesos y parámetros para mejorar su capacidad de hacer predicciones precisas. Usar más épocas puede permitir que el modelo aprenda patrones más complejos en los datos, pero también aumenta el riesgo de sobreajuste (overfitting) si no se monitorea adecuadamente (Agustian et al., 2021).

- **Tamaño del lote:** El tamaño del lote se refiere a la cantidad de ejemplos de entrenamiento que se utilizan en cada paso de actualización de los pesos del modelo. En el entrenamiento de una red neuronal,

en lugar de actualizar los pesos después de cada imagen, se actualizan después de procesar un conjunto de imágenes (el lote). Un tamaño de lote más grande puede acelerar el entrenamiento, ya que el procesamiento en paralelo es más eficiente en hardware moderno. Sin embargo, puede requerir más memoria. Un tamaño de lote más pequeño puede ser útil si tienes limitaciones de memoria y también puede ayudar a mejorar la convergencia del modelo (Agustian et al., 2021).

- **Tasa de aprendizaje:** La tasa de aprendizaje es un hiperparámetro que determina cuánto deben ajustarse los pesos del modelo en cada paso de actualización durante el entrenamiento. Una tasa de aprendizaje baja significa que los pesos se actualizan en pequeñas cantidades, lo que puede llevar a un entrenamiento más lento, pero más preciso. Una tasa de aprendizaje alta permite actualizaciones más grandes y, a menudo, un entrenamiento más rápido, pero puede hacer que el modelo no converja correctamente. Elegir la tasa de aprendizaje adecuada es crítico, y a menudo se requiere ajuste y experimentación para encontrar el valor óptimo (Artola Moreno, 2019).

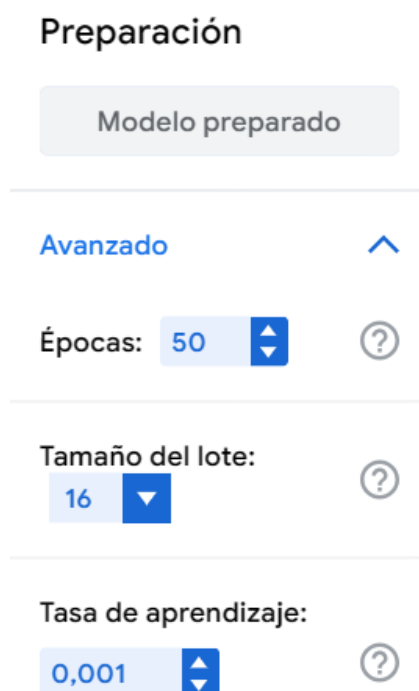
3. RESULTADOS Y DISCUSIÓN

3.1. Resultados

3.1.1. Primer Dataset

En la tabla 1 se pueden ver los valores obtenidos de las pruebas realizadas al dataset1 el cual tiene un conjunto de entrenamiento de 10 imágenes, 2 imágenes para validación y 5 imágenes para las pruebas. El modelo fue entrenado bajo las siguientes características, como se aprecia en la ilustración 2:

- Épocas: 50
- Tamaño del lote: 16
- Tasa de aprendizaje: 0.001.



The image shows a 'Preparación' (Preparation) interface for a model. At the top, there is a button labeled 'Modelo preparado'. Below this, there is a section titled 'Avanzado' (Advanced) with an upward-pointing arrow. The parameters are listed as follows:

- Épocas:** 50 (with up and down arrows and a help icon)
- Tamaño del lote:** 16 (with a dropdown arrow and a help icon)
- Tasa de aprendizaje:** 0,001 (with up and down arrows and a help icon)

Ilustración 2 Entrenamiento Dataset 1 (10 imágenes)

Una vez realizado el entrenamiento del modelo, se realizó las pruebas correspondientes y conseguimos los siguientes grados de similitud para cada clase:

	Clases	Prueba 1	Prueba 2	Prueba 3	Prueba 4	Prueba 5
1	Iguera	0,74	0,78	0,83	0,87	0,73
2	Rufino	0,54	0,65	0,87	0,97	0,62
3	Blacki	0,98	0,74	0,59	0,78	0,78
4	Luka	0,68	0,82	0,73	0,89	0,74
5	Bodoko	0,54	0,47	0,93	0,78	0,83
6	Bombon	0,74	0,56	0,87	0,56	0,67
7	Durazno	0,74	0,52	0,87	0,91	0,79
8	Max	0,63	0,72	0,68	0,74	0,56
9	Kiara	0,58	0,52	0,87	0,74	0,69
10	Totoro	0,85	0,74	0,68	0,76	0,91

Tabla 1 Pruebas Dataset 1

3.1.2. Segundo Dataset

En la tabla 2 se pueden ver los valores obtenidos de las pruebas realizadas al dataset2 el cual tiene un conjunto de entrenamiento de 30 imágenes, 5 imágenes para validación y 5 imágenes para las pruebas. El modelo fue entrenado bajo las siguientes características, lo que también se visualiza en la ilustración 3:

- Épocas: 50
- Tamaño del lote: 16
- Tasa de aprendizaje: 0.001.

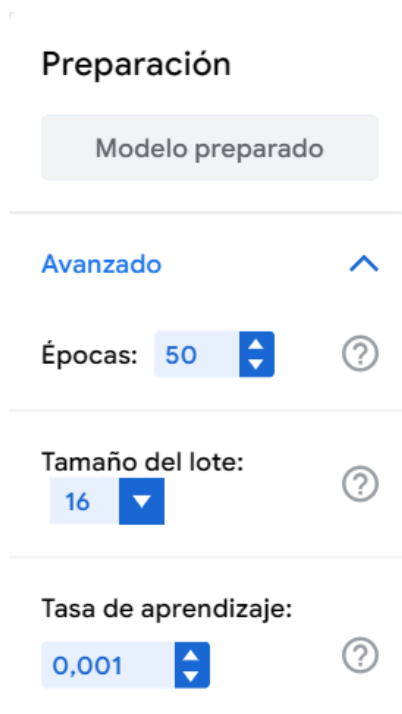


Ilustración 3 Entrenamiento Dataset 2 (30 imágenes)

Una vez realizado el entrenamiento del modelo, se realizó las pruebas correspondientes y conseguimos los siguientes grados de similitud para cada clase:

	Clases	Prueba 1	Prueba 2	Prueba 3	Prueba 4	Prueba 5
1	Iguera	0,78	0,87	0,88	0,78	0,79
2	Rufino	0,69	0,72	0,86	0,79	0,83
3	Blacki	0,98	0,94	0,91	0,83	0,84

4	Luka	0,76	0,86	0,79	0,93	0,8
5	Bodoko	0,66	0,53	0,96	0,84	0,87
6	Bombon	0,8	0,63	0,93	0,6	0,77
7	Durazno	0,77	0,69	0,74	0,91	0,97
8	Max	0,56	0,75	0,71	0,77	0,9
9	Kiara	0,61	0,68	0,78	0,74	0,71
10	Totoro	0,88	0,77	0,71	0,73	0,87

Tabla 2 Pruebas Dataset 2

3.1.3. Tercer Dataset

En la tabla 3 se pueden ver los valores obtenidos de las pruebas realizadas al dataset2 el cual tiene un conjunto de entrenamiento de 50 imágenes, 8 imágenes para validación y 5 imágenes para las pruebas. El modelo fue entrenado bajo las siguientes características; tal como se muestra en la ilustración 4:

- Épocas: 50
- Tamaño del lote: 16
- Tasa de aprendizaje: 0.001.

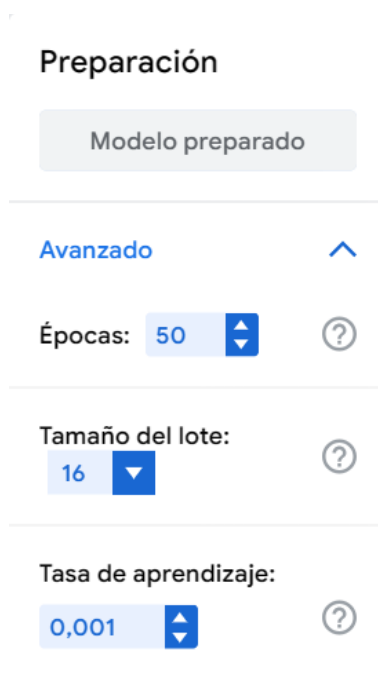


Ilustración 4 Entrenamiento Dataset 3 (50 imágenes)

Una vez realizado el entrenamiento del modelo, se realizó las pruebas correspondientes y conseguimos los siguientes grados de similitud para cada clase:

	Clases	Prueba 1	Prueba 2	Prueba 3	Prueba 4	Prueba 5
1	Iguera	0,93	0,95	0,98	0,89	0,94
2	Rufino	0,96	0,93	0,89	0,97	0,95
3	Blacki	0,98	0,94	0,95	0,87	0,96
4	Luka	0,97	0,99	0,97	0,93	0,99
5	Bodoko	0,96	0,88	0,96	0,89	0,93
6	Bombon	0,99	0,97	0,96	0,9	0,87
7	Durazno	0,9	0,87	0,83	0,99	0,95
8	Max	0,87	0,86	0,91	0,98	0,9
9	Kiara	0,89	0,86	0,87	0,9	0,98
10	Totoro	0,99	0,99	0,97	0,94	0,93

Tabla 3 Pruebas Dataset 3

3.1.4. Prueba de normalidad de los datos

Para determinar si los datos siguen una distribución normal se aplicó la prueba de Shapiro-Wilk, se utilizó esta prueba de normalidad ya que nuestros datos son menores a 50. Se plantearon las siguientes hipótesis:

H0: La variable presenta una distribución normal

H1: La variable presenta una distribución no normal

Toma de decisión:

Sig(p valor) > alfa: No rechazar H0 (normal).

Sig(p valor) < alfa: Rechazar H0 (no normal)

Donde alfa representa la significancia, que es igual al 5% (0,05).

3.1.4.1. Prueba 1 de normalidad

Aplicando la prueba de Shapiro-Wilk en el software SPSS se obtuvo el siguiente resultado, expresado en la ilustración 5:

Shapiro-Wilk		
Estadístico	gl	Sig.
,889	10	,165
,977	10	,946
,927	10	,422

Ilustración 3 Prueba grupo 1 de normalidad

Para los tres Datasets de prueba se obtuvieron valores de p de 0.165, 0.946 y 0.422. Dichos valores son mayores al valor del alfa de 0.05 por lo que no se rechaza la hipótesis nula concluyendo que los datos presentan una distribución normal.

3.1.4.2. Prueba 2 de normalidad

Aplicando la prueba de Shapiro-Wilk en el software SPSS se obtuvo el siguiente resultado, tal como se muestra en la ilustración 6:

Shapiro-Wilk		
Estadístico	gl	Sig.
,879	10	,128
,981	10	,972
,910	10	,281

Ilustración 6 Prueba grupo 2 de normalidad

Para los tres Datasets de prueba se obtuvieron valores de p de 0.128, 0.972 y 0.281. Dichos valores son mayores al valor del alfa de 0.05 por lo que no se rechazó la hipótesis nula concluyendo que los datos presentan una distribución normal.

3.1.4.3. Prueba 3 de normalidad

Aplicando la prueba de Shapiro-Wilk en el software SPSS se obtuvo el siguiente resultado, el cual se muestra en la ilustración 7:

Shapiro-Wilk		
Estadístico	gl	Sig.
,866	10	,090
,921	10	,364
,878	10	,123

Ilustración 7 Prueba grupo 3 de normalidad

Para los tres Datasets de prueba se obtuvieron valores de p de 0.090, 0.364 y 0.123. Dichos valores son mayores al valor del alfa de 0.05 por lo que no se rechazó la hipótesis nula concluyendo que los datos presentan una distribución normal.

3.1.4.4. Prueba 4 de normalidad

Aplicando la prueba de Shapiro-Wilk en el software SPSS se obtuvo el siguiente resultado, mostrado en la ilustración 8:

Shapiro-Wilk		
Estadístico	gl	Sig.
,912	10	,297
,955	10	,730
,936	10	,509

Ilustración 4 Prueba grupo 4 de normalidad

Para los tres Datasets de prueba se obtuvieron valores de p de 0.297, 0.730 y 0.509. Dichos valores son mayores al valor del alfa de 0.05 por lo que no rechazamos la hipótesis nula concluyendo que los datos presentan una distribución normal.

3.1.4.5. Prueba 5 de normalidad

Aplicando la prueba de Shapiro-Wilk en el software SPSS se obtuvo el siguiente resultado, el cual se muestra en la ilustración 9:

Shapiro-Wilk		
Estadístico	gl	Sig.
,957	10	,751
,990	10	,996
,995	10	1,000

Ilustración 9 Prueba grupo 5 de normalidad

Para los tres Datasets de prueba se obtuvieron valores de p de 0.751, 0.996 y 1.00 Dichos valores son mayores al valor del alfa de 0.05 por lo que no rechazamos la hipótesis nula concluyendo que los datos presentan una distribución normal. Todos los datos están distribuidos de manera normal, lo que nos indica que se puede trabajar con pruebas paramétricas.

3.1.5. Análisis de precisión

Se seleccionó ANOVA para esta prueba porque se va a comparar la precisión de más de dos grupos. Planteando las siguientes hipótesis:

H0: Las medias de los grupos son iguales

H1: Al menos una de las medias de los grupos es diferente

Si el grado de significancia es $>$ al 0.05 se acepta la hipótesis nula, caso contrario se acepta la hipótesis alternativa. Ingresando todos los datos obtenidos en las pruebas de los tres grupos y con un nivel de confianza del 95% obtuvimos los siguientes resultados, mostrados en la ilustración 10:

ANOVA					
precision	Suma de cuadrados	gl	Media cuadrática	F	Sig.
Entre grupos	1,037	2	,518	52,431	<,001
Dentro de grupos	1,453	147	,010		
Total	2,490	149			

Ilustración 10 Estadístico ANOVA de los tres grupos

Como se puede ver en la ilustración 11, el grado de significancia es menor al 0.001 por lo que se concluye que al menos una de las medias de los grupos es diferente.

Gráficos de medias

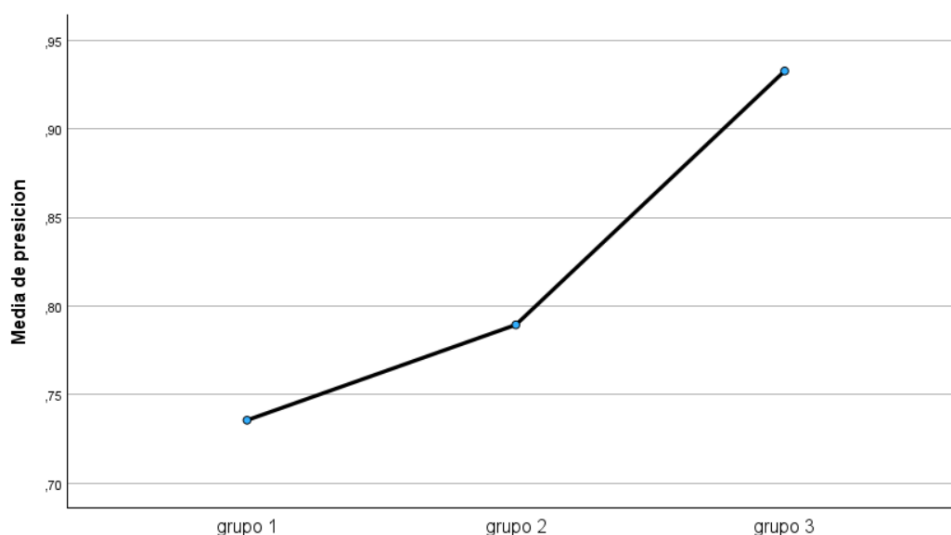


Ilustración 11 Gráfica de medias de los tres grupos

Como se ve en el gráfico las medias de precisión varían significativamente entre los grupos concluyendo que mientras más imágenes se le provea al modelo para su entrenamiento y validación, mejora su grado de precisión.

3.1.5.1. Análisis de estabilidad de precisión en función del tamaño del conjunto de datos de entrenamiento

Para realizar este análisis se tomaron pruebas de precisión con un grupo adicional llamado grupo 4, dicho grupo está compuesto por un Dataset de 70 imágenes para el entrenamiento y 11 para validación

entrenado bajo los mismos parámetros de los demás grupos que son 50 épocas, un tamaño del lote de 16 y una tasa de aprendizaje de 0.001 del cual se obtuvieron los siguientes datos, representados en la tabla 4:

	Clases	Prueba 1	Prueba 2	Prueba 3	Prueba 4	Prueba 5
1	Iguera	0,94	0,96	0,99	0,9	0,95
2	Rufino	0,97	0,94	0,9	0,98	0,96
3	Blacki	0,99	0,95	0,96	0,88	0,97
4	Luka	0,98	0,98	0,98	0,94	0,97
5	Bodoko	0,97	0,89	0,97	0,9	0,94
6	Bombon	0,98	0,98	0,97	0,91	0,88
7	Durazno	0,91	0,88	0,84	0,97	0,96
8	Max	0,88	0,87	0,92	0,99	0,91
9	Kiara	0,9	0,87	0,88	0,91	0,99
10	Totoro	0,98	0,98	0,98	0,95	0,94

Tabla 4 Datos análisis de estabilidad de precisión

Se utilizó T de student para realizar esta prueba ya que se van a comparar los grupos 3 y 4, planteando las siguientes hipótesis:

H0: No hay diferencia significativa en la precisión del modelo entre los diferentes tamaños de conjuntos de datos de entrenamiento.

H1: Existe una diferencia significativa en la precisión del modelo entre al menos dos de los tamaños de conjuntos de datos de entrenamiento.

Aplicando el estadístico T con un grado de confianza del 95% obtuvimos los siguientes resultados, que se muestran en la ilustración 12.

Prueba de Levene de igualdad de varianzas				prueba t para la igualdad de medias							
		F	Sig.	t	gl	P de un factor	Significación P de dos factores	Diferencia de medias	Diferencia de error estándar	95% de intervalo de confianza de la diferencia	
										Inferior	Superior
precision	Se asumen varianzas iguales	,353	,554	-.841	98	,201	,403	-.00720	,00856	-.02419	,00979
	No se asumen varianzas iguales			-.841	97,485	,201	,403	-.00720	,00856	-.02419	,00979

Ilustración 12 Prueba T de student comparación de la precisión de dos grupos

El valor (p) obtenido es de 0.403, el nivel de significancia (α) es del 0.05 concluyendo que como el valor de p es mayor que el valor de alfa ($0.403 > 0.05$) aceptamos la hipótesis nula por lo cual no existe una diferencia significativa en la precisión del modelo entre los diferentes tamaños de conjuntos de datos de entrenamiento de 50 imágenes y 70 imágenes, por lo que se podrían utilizar un número de imágenes en un grupo de entre 50 a 70.

3.1.6. Evaluación de la Exactitud y Precisión por Clase

Para evaluar la precisión y la exactitud por clase se utilizó un Dataset de 60 imágenes para el entrenamiento, 9 para validación y 5 para realización de pruebas.

- Precisión por clase: es una métrica utilizada para evaluar el rendimiento de un modelo de aprendizaje automático en la clasificación de diferentes clases. Esta métrica proporciona información sobre la capacidad del modelo para realizar predicciones precisas en cada clase individualmente [5]. TeachableMachine nos arroja los siguientes resultados, que se muestran en la ilustración 13:

Precisión por clase

CLASS	ACCURACY	# SAMPLES
Iguera	1.00	9
Rufino	1.00	9
Blacki	1.00	9
Luka	1.00	9
Bodoko	1.00	9
Bombon	1.00	9
Durazno	1.00	9
Max	1.00	9
Kiara	1.00	9
Totoro	1.00	9

Ilustración 13 Precisión por clase

• Matriz de confusiones: es una herramienta que muestra cómo el modelo de clasificación ha realizado sus predicciones en comparación con las etiquetas verdaderas de los datos. En otras palabras, indica cuántas veces el modelo ha clasificado las muestras en las clases correctas y cuántas veces ha cometido errores al clasificarlas en clases incorrectas; la misma se encuentra representada en la ilustración 14.

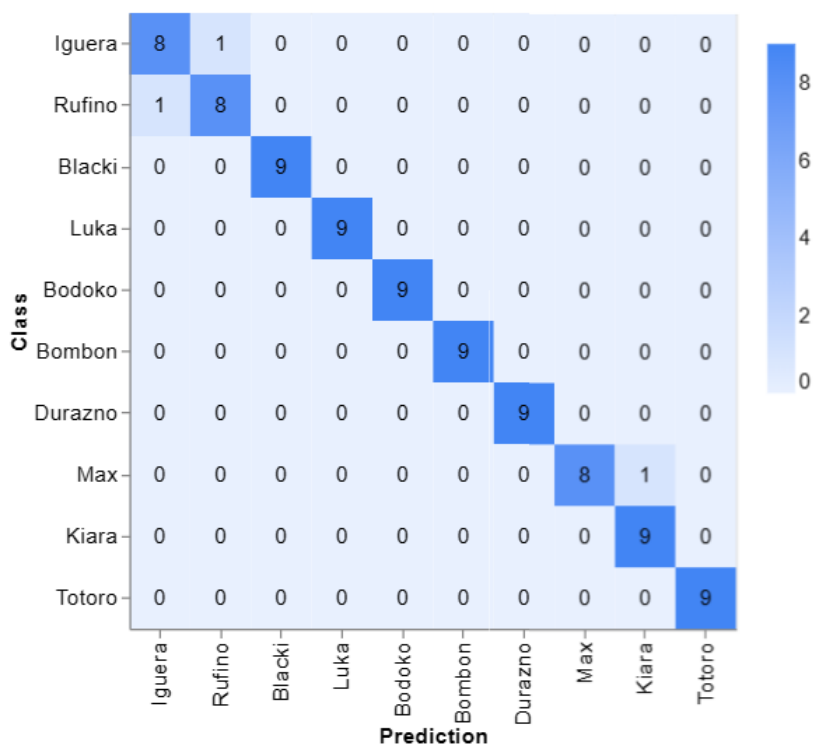


Ilustración 14 Matriz de confusiones

• Precisión por épocas: se refiere a un seguimiento del desempeño de un modelo de aprendizaje automático en diferentes momentos durante el proceso de entrenamiento, generalmente en un gráfico o

tabla que muestra la precisión del modelo en un conjunto de datos de validación en cada época. La precisión en este contexto se relaciona con la capacidad del modelo para hacer predicciones correctas; lo cual se detalla en la ilustración 15.

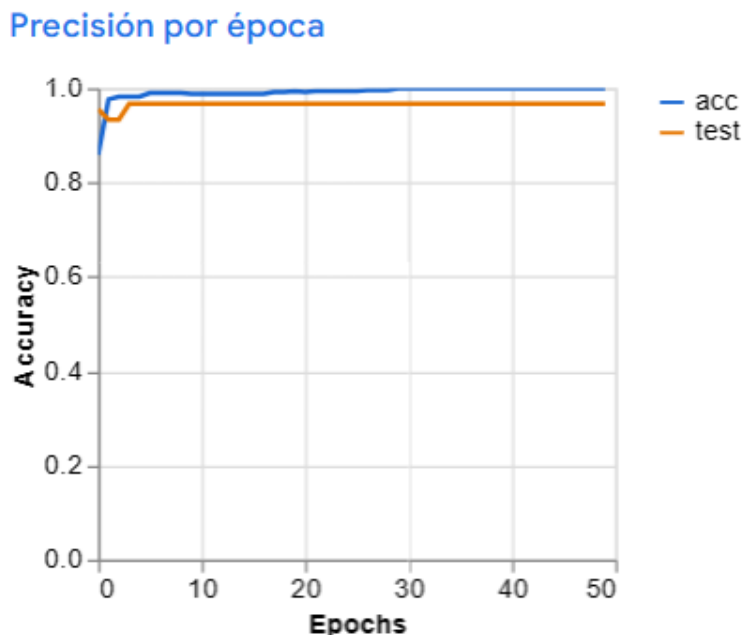


Ilustración 15 Precisión por épocas

- Pérdida por épocas: representa una medida de cuán equivocadas son las predicciones del modelo en comparación con las etiquetas verdaderas de los datos de entrenamiento. En otras palabras, es una medida de cuán lejos está el modelo de hacer predicciones perfectas. La idea es minimizar esta pérdida durante el entrenamiento para que el modelo haga predicciones más precisas, mostradas en la ilustración 16.

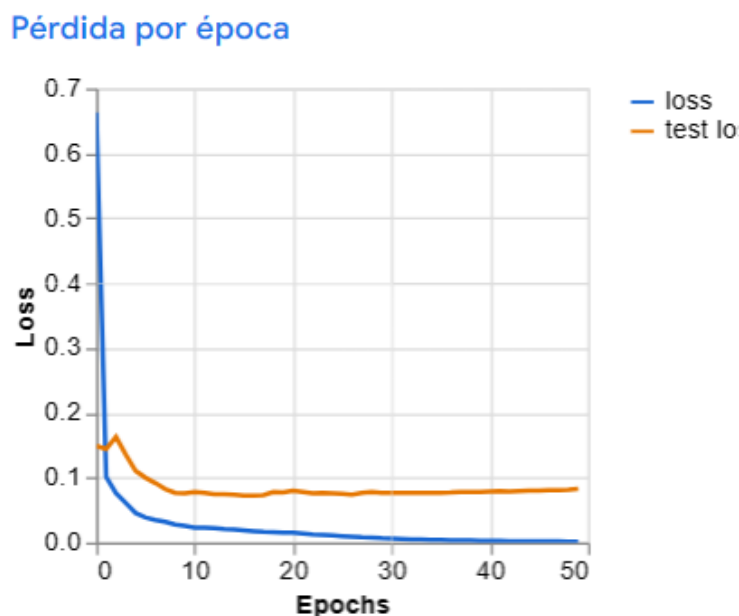


Ilustración 16 Pérdida por épocas

Una vez obtenido los resultados anteriores, se va a realizar los cálculos para obtener la precisión global y por clase, además de la exactitud y la sensibilidad donde se tiene:

TP = Verdaderos positivos

TN= Verdaderos negativos

FP= Falsos positivos

FN= Falsos negativos

- Precisión global: La precisión global del modelo se calcula mediante la siguiente fórmula:

$$\frac{\textit{suma de diagonal principal}}{TP + TN + FP + FN}$$

$$\frac{8 + 8 + 9 + 9 + 9 + 9 + 9 + 8 + 9 + 9}{8 + 1 + 1 + 8 + 9 + 9 + 9 + 9 + 9 + 9 + 8 + 1 + 9 + 9}$$

Dándonos como resultado:

$$\frac{87}{90}$$

O un 96.66% de precisión global

- Precisión por clase: para obtener la precisión por cada clase se utiliza la siguiente formula:

$$\frac{8}{8 + 1} = 0.88$$

- Precisión de la clase Rufino

$$\frac{8}{8 + 1} = 0.88$$

- Precisión de la clase Blaki

$$\frac{9}{9} = 1$$

- Precisión de la clase Luka

$$\frac{9}{9} = 1$$

- Precisión de la clase Bodoko

$$\frac{9}{9} = 1$$

- Precisión de la clase Durazno

$$\frac{9}{9} = 1$$

- Precisión de la clase Max

$$\frac{8}{8 + 1} = 0.88$$

- Precisión de la clase Kiara

$$\frac{9}{9} = 1$$

- Precisión de la clase Totoro

$$\frac{9}{9} = 1$$

• Recall: mide la capacidad del modelo para identificar correctamente todos los ejemplos positivos de una clase específica, tal como se muestra en la ilustración 17.

$$\frac{TP}{TP + FN}$$

- Recall de la clase Iguera

$$\frac{8}{8 + 2} = 0.8$$

- Recall de la clase Rufino

$$\frac{8}{8 + 2} = 0.8$$

- Recall de la clase Blaki

$$\frac{9}{9} = 1$$

- Recall de la clase Luka

$$\frac{9}{9} = 1$$

- Recall de la clase Bodoko

$$\frac{9}{9} = 1$$

- Recall de la clase Bombon

$$\frac{9}{9} = 1$$

- Recall de la clase Durazno

$$\frac{9}{9} = 1$$

- Recall de la clase Max

$$\frac{8}{8 + 1} = 0.88$$

- Recall de la clase Kiara

$$\frac{9}{9} = 1$$

- Recall de la clase Totoro

$$\frac{9}{9} = 1$$

3.2. Discusión

Al examinar los distintos conjuntos de datos, como se muestra en las ilustraciones 5, 6, 7, 8 y 9, se llega a la conclusión de que los datos exhiben una distribución normal. Esto se evidencia al observar que los valores de p obtenidos son superiores al nivel alfa establecido de 0.05. Este hallazgo respalda la

afirmación de que los datos siguen una distribución normal. En consecuencia, se confirma la idoneidad de aplicar pruebas paramétricas en el análisis de estos datos.

Al evaluar la precisión mediante el análisis de varianza (ANOVA) para comparar las medias de los tres grupos, según se evidencia en la ilustración 10, se observa que el valor de p obtenido es inferior a 0.05. Este resultado indica que al menos una de las medias de los grupos difiere significativamente. La confirmación de esta variación en las medias se refleja en la ilustración 11, lo que permite concluir que la precisión varía de manera significativa entre los grupos. Este hallazgo respalda la noción de que proporcionar al modelo un mayor número de imágenes para su entrenamiento y validación resulta en una mejora significativa en su precisión.

Al examinar la precisión en relación con el tamaño del conjunto de datos de entrenamiento de los grupos 3 y 4, que contienen 50 y 70 imágenes respectivamente, se llegó a la conclusión de que el valor de p obtenido es de 0.403, mientras que el nivel de significancia (α) es del 0.05. Al comparar estos valores, se observa que el valor de p (0.403) es mayor que el valor de alfa (0.05) como se puede ver en la ilustración 12. En consecuencia, se concluye que no existe una diferencia significativa en la precisión del modelo entre los diferentes tamaños de conjuntos de datos de entrenamiento, específicamente entre aquellos con 50 y 70 imágenes. Esto sugiere que se podría utilizar un número de imágenes en un grupo en el rango de 50 a 70 sin impacto significativo en la precisión del modelo.

Al examinar la fórmula 1 para calcular la precisión global, se concluye que el modelo exhibe una precisión global del 96.66%. Sin embargo, al analizar la precisión en las clases específicas de Iguera, Rufino y Max, se observa que se obtuvo una precisión del 88%. Este resultado se atribuye a la presencia de factores que impactaron la precisión de estas clases, dado que se registraron falsos positivos que comprometieron ligeramente la precisión en estas categorías. Por otro lado, destacamos que se logró una precisión del 100% en las demás clases.

En relación con el recall por clase, que evalúa la capacidad del modelo para identificar correctamente todos los ejemplos positivos de una clase específica, se obtuvieron tasas de 80% en las clases Iguera y Rufino, y un 88% en la clase Max. En contraste, se alcanzó un recall del 100% en las restantes clases. Estos resultados subrayan la importancia de considerar no solo la precisión global, sino también la precisión y el recall específicos por clase para una evaluación más detallada del rendimiento del modelo.

CONCLUSIONES

Al examinar las ilustraciones 5, 6, 7, 8 y 9, se determinó que los datos presentan una distribución normal. Esta conclusión se respalda mediante la observación de valores de p superiores al nivel alfa establecido de 0.05. En consecuencia, se confirma la idoneidad de aplicar pruebas paramétricas para el análisis de estos datos.

La aplicación del análisis de varianza (ANOVA) reveló que, al comparar las medias de los tres grupos (ilustración 10), al menos una de las medias difiere significativamente. La ilustración 11 refleja esta variación en las medias, concluyendo que la precisión varía de manera significativa entre los grupos. Esto

respalda la idea de que proporcionar al modelo un mayor número de imágenes para su entrenamiento y validación mejora significativamente su precisión.

La evaluación de la precisión en relación con el tamaño del conjunto de datos de entrenamiento (grupos 3 y 4 con 50 y 70 imágenes, respectivamente) reveló un valor de p de 0.403, superior al nivel de significancia α de 0.05 (ilustración 12). Esta conclusión indica que no existe una diferencia significativa en la precisión del modelo entre los conjuntos de datos de 50 y 70 imágenes. Por lo tanto, se sugiere que se podrían utilizar entre 50 y 70 imágenes en un grupo sin afectar significativamente la precisión del modelo.

Al analizar la precisión por clase, se observó una precisión global del 96.66%. Sin embargo, se identificaron desafíos específicos en las clases de Iguera, Rufino y Max, con una precisión del 88%. Esto se atribuye a la presencia de falsos positivos que afectaron ligeramente la precisión en estas clases. A pesar de esto, se logró una precisión del 100% en las demás clases.

El recall por clase reveló tasas del 80% en las clases Iguera y Rufino, y un 88% en la clase Max. En contraste, se logró un recall del 100% en las demás clases. Estos resultados subrayan la importancia de considerar tanto la precisión global como la precisión y recall específicos por clase para obtener una evaluación completa del rendimiento del modelo.

REFERENCIAS

- Agustian, D., Pertama, P. P. G. P., Crisnapati, P. N., & Novayanti, P. D. (2021). *Implementation of Machine Learning Using Google's Teachable Machine Based on Android*. 1-7. <https://doi.org/10.1109/ICORIS52787.2021.9649528>
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of big Data*, 8(1), 53. <https://doi.org/10.1186/s40537-021-00444-8>
- Andreae, P. M., & Andreae, J. H. (1978). A teachable machine in the real world. *International Journal of Man-Machine Studies*, 10(3), 301-312. [https://doi.org/10.1016/S0020-7373\(78\)80048-0](https://doi.org/10.1016/S0020-7373(78)80048-0)
- Artola Moreno, Á. (2019). *Clasificación de imágenes usando redes neuronales convolucionales en Python* [Trabajo Fin de Grado, Universidad de Sevilla]. <https://idus.us.es/server/api/core/bitstreams/c22e9088-218a-4554-9b27-6c0ebf89d0e8/content>
- Arulprakash, E., & Aruldoss, M. (2022). A study on generic object detection with emphasis on future research directions. *Journal of King Saud University-Computer and Information Sciences*, 34(9), 7347-7365. <https://doi.org/10.1016/j.jksuci.2021.08.001>
- Bodero, E. M., Lopez, M. P., Congacha, A. E., Cajamarca, E. E., & Morales, C. H. (2020). Google Colaboratory como alternativa para el procesamiento de una red neuronal convolucional. *Revista Espacios*, 41(07).
- Carney, M., Webster, B., Alvarado, I., Phillips, K., Howell, N., Griffith, J., Jongejan, J., Pitaru, A., & Chen, A. (2020). *Teachable machine: Approachable Web-based tool for exploring machine learning classification*. 1-8. <https://doi.org/10.1145/3334480.3382839>
- Kacorri, H. (2017). Teachable machines for accessibility. *ACM SIGACCESS accessibility and computing*, 119, 10-18. <https://doi.org/10.1145/3167902.316790>
- Liu, X., Soh, K. G., Dev Omar Dev, R., Li, W., & Yi, Q. (2023). Design and implementation of adolescent health Latin dance teaching system under artificial intelligence technology. *Plos one*, 18(11), e0293313. <https://doi.org/10.1371/journal.pone.0293313>
- Massiris, M., Delrieux, C., & Fernández Muñoz, J. Á. (2018). Detección de equipos de protección personal mediante red neuronal convolucional YOLO. *Jornadas de Automática, Badajoz*, 5(7), 1022-1029. <https://doi.org/10.17979/spudc.9788497497565.1022>
- O'shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *Neural and Evolutionary Computing*. <https://doi.org/10.48550/arXiv.1511.08458>
- Prasad, R., & Manjunath, T. (2024). AI-ML Trained Object Recognition System Development using Google Teachable Machine with the Help of Data Sciences. *Grenze International Journal of Engineering & Technology (GIJET)*, 10.

- Saqib, S. M., Iqbal, M., Mazhar, T., Shahzad, T., Ouahada, K., & Hamam, H. (2025). Effectiveness of Teachable Machine, mobile net, and YOLO for object detection: A comparative study on practical applications. *Egyptian Informatics Journal*, 30, 100680. <https://doi.org/10.1016/j.eij.2025.100680>
- Sharma, T., Debaque, B., Duclos, N., Chehri, A., Kinder, B., & Fortier, P. (2022). Deep learning-based object detection and scene perception under bad weather conditions. *Electronics*, 11(4), 563. <https://doi.org/10.3390/electronics11040563>
- Siddiqui, N. (2023). Creating Deep Convolutional Neural Networks for Image Classification. *The Programming Historian*. <https://doi.org/10.46430/phen0108>
- Tjaden, J., & Tjaden, B. (2023). MLpronto: A tool for democratizing machine learning. *Plos one*, 18(11), e0294924. <https://doi.org/10.1371/journal.pone.0294924>
- Trujillo Santillán, N. S., Grijalva Rosero, C. J., & Herrera Mendoza, B. R. (2024). Estimación de la población de perros y gatos en situación de calle dentro del cantón Riobamba -Ecuador. *RECIENA Edición Especial*, 4(1), 27-32. <https://doi.org/10.47187/tfw9et71>
- Zhou, M., Fung, I., Yang, L., Wan, N., Di, K., & Wang, T. (2023). LostNet: A smart way for lost and find. *arXiv preprint arXiv:2301.02277*. <https://doi.org/10.48550/arXiv.2301.02277>