

Detección de imágenes o vídeos modificados mediante Redes Neuronales

Miguel del Arco Marquez

Resum– Resum del projecte, màxim 10 línies.
.....
.....
.....
.....
.....
.....
.....
.....
.....

Palabras claves– Aprendizaje profundo, Detección de imágenes modificadas, Conjuntos de datos, Preprocesamiento de imágenes, Detección de manipulación específica.

Abstract– Versió en anglès del resum
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

Keywords– Deep learning, Detection of modified images, Data sets, Image preprocessing, Detection of specific manipulation.



gencia artificial.

Para abordar esta problemática, se han desarrollado diversas técnicas basadas en el aprendizaje profundo que permiten detectar si una imagen ha sido modificada o no. El aprendizaje profundo, también conocido como deep learning, es una rama de la inteligencia artificial que utiliza redes neuronales artificiales para aprender y realizar tareas complejas, como el reconocimiento de objetos y el procesamiento de lenguaje natural.

En este trabajo, se abordará la detección de imágenes modificadas mediante el uso de técnicas de aprendizaje profundo. Se discutirán los diferentes modelos de deep learning que se pueden utilizar para este fin, así como los pasos necesarios para llevar a cabo el entrenamiento y la evaluación de los modelos. Asimismo, se destacará la importancia de contar con un conjunto de datos etiquetado y representativo para el entrenamiento de los modelos, así como la necesi-

1 INTRODUCCIÓN - CONTEXTO DEL TRABAJO

EN la actualidad, el uso de imágenes en la comunicación y en la toma de decisiones es cada vez más común. Sin embargo, existe la posibilidad de que algunas de estas imágenes sean manipuladas o modificadas para engañar o tergiversar la información que se presenta. Por esta razón, la detección de imágenes modificadas se ha convertido en una tarea importante en el campo de la inteli-

- E-mail de contacto: migueldemollet10@gmail.com
- Mención realizada: Computación.
- Trabajo tutorizado por: Jordi Serra Ruiz (Departamento de Ciencias de la Computación)
- Curs 2022/23

dad de ajustar y actualizar periódicamente el modelo para mejorar su rendimiento.

En definitiva, este trabajo tiene como objetivo brindar una visión general de cómo se puede utilizar el aprendizaje profundo para detectar imágenes modificadas y destacar la relevancia de esta tarea en la actualidad.

2 OBJETIVOS

Los objetivos que supone un trabajo de esta magnitud son los siguientes.

1. Identificar y evaluar los modelos de aprendizaje profundo más efectivos para detectar imágenes modificadas en un conjunto de datos específico.
2. Implementar y entrenar un modelo de aprendizaje profundo para la detección de imágenes modificadas, utilizando un conjunto de datos representativo.
3. Analizar la efectividad del modelo en la detección de diferentes tipos de modificaciones, como la manipulación de la información visual, la eliminación de objetos o la inserción de objetos.
4. Comparar el rendimiento del modelo de aprendizaje profundo con otras técnicas de detección de imágenes modificadas, como el análisis forense de imágenes o la detección de patrones.
5. Proporcionar recomendaciones para mejorar la precisión y la eficacia del modelo de aprendizaje profundo en la detección de imágenes modificadas, como la incorporación de nuevos datos de entrenamiento o la adaptación del modelo a nuevos tipos de modificaciones.

3 METODOLOGÍA

Durante el desarrollo del proyecto, se utilizará GitHub como herramienta de gestión de versiones y almacenamiento del código. Esto permitirá tener un seguimiento detallado del progreso del trabajo y una trazabilidad adecuada de los cambios realizados. Además, se ha establecido un proceso de revisión regular con una frecuencia de 15 días, en el cual se compartirá el avance del trabajo con el tutor designado, Jordi Serra Ruiz, del departamento de Ciencias de la Computación, y se recibirá feedback para asegurar que se está avanzando adecuadamente hacia los objetivos del TFG. Este enfoque garantizará una comunicación fluida entre el estudiante y el tutor, lo que resultará en un trabajo de alta calidad.

4 ESTADO DEL ARTE

El campo de la detección de imágenes modificadas mediante el uso de técnicas de aprendizaje profundo se encuentra en constante evolución y desarrollo. A continuación, se presentan algunos de los avances más relevantes y actuales en esta área:

- **Herramientas de pago:** Existe una herramienta que podemos encontrar en la suite de *Adobe Analytics*[1][2],

esta pretende detectar si una imagen ha sido modificada mediante algoritmos de Deep Learning.

Esta tecnología permite a los creadores de contenido proteger sus derechos de autor y verificar la autenticidad de la imagen. Por otro lado permite detectar si dicha imagen es original o ha sido modificada.

Para tener una mejor idea de como esta funciona, podemos observar como en la figura 1 esta es capaz de detectar objetos añadidos a imágenes.

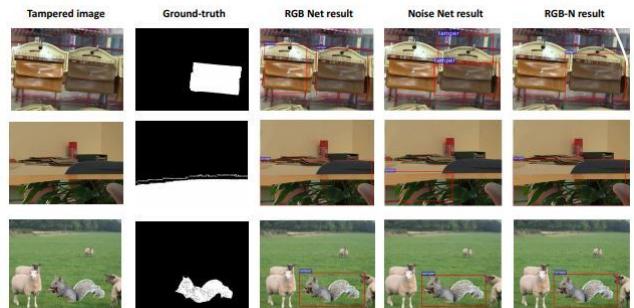


Fig. 1: Ejemplo de funcionamiento de la IA de Adobe

- **Redes Neuronales Convolucionales (CNN):** Las redes neuronales convolucionales son el modelo de deep learning más utilizado en la detección de imágenes modificadas, y han demostrado tener un alto nivel de precisión en la detección de imágenes modificadas por scripts de *Adobe Photoshop®*.

Podemos ver trabajos ya realizados en el cual usan este tipo de red neuronal, un caso sería el trabajo de **Detecting Photoshopped Faces by Scripting Photoshop**[3]

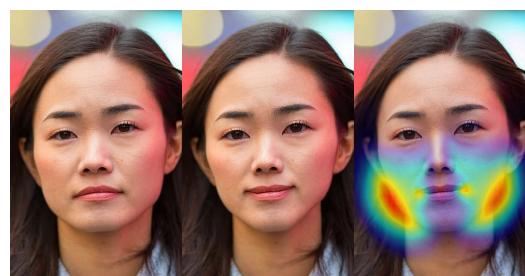


Fig. 2: Resultados Detecting Photoshopped Faces by Scripting Photoshop

Para entender mejor la figura 2 debemos de ver que esta imagen esta compuesta compuesta por 3 sub-imagenes de izquierda a derecha seria lo siguiente.

1. Imagen original
2. Imagen modificada
3. Output del modelo

- **Uso de tecnologías existentes como Deep Fake:** Es una técnica de inteligencia artificial que se utiliza para crear videos o imágenes manipuladas que parecen ser auténticas, pero en realidad son falsas. Esta técnica utiliza algoritmos de aprendizaje profundo para entrenar modelos de redes neuronales que pueden analizar,

sintetizar y manipular el contenido de una imagen o un video. Con esta técnica, es posible crear videos y fotos que parecen ser reales, pero que en realidad son el resultado de la manipulación de contenido existente, como el rostro de una persona. Los deepfakes se han utilizado en algunos casos para crear noticias falsas, difamar a personas o para fines de entretenimiento. Es importante destacar que los deepfakes pueden ser utilizados de manera engañosa, por lo que es importante ser cautelosos al consumir contenido generado por esta técnica [4].

Para entender mejor esta técnica podemos ver en la figura 3 el resultado de aplicar esta técnica.



Fig. 3: Resultado del Deep Fake (Output—Input)

Existen papers como **FaceForensics++ - Learning to Detect Manipulated Facial Images**[5] en el cual trata sobre la preocupación creciente acerca de la capacidad de generar y manipular imágenes sintéticas con un alto grado de realismo, lo que puede tener implicaciones graves en la sociedad, ya que puede conducir a una pérdida de confianza en el contenido digital y difundir información falsa o noticias falsas.

Este propone la detección de manipulaciones faciales y da un benchmark automatizado para la evaluación de la eficacia de los métodos de detección.

El benchmark se basa en representantes destacados de manipulaciones faciales y contiene una base de datos de más de 1.8 millones de imágenes manipuladas. Los autores muestran que el uso de conocimiento específico del dominio mejora significativamente la detección de falsificaciones y supera claramente a los observadores humanos. En resumen, el trabajo aborda un problema importante en el ámbito de la generación y manipulación de imágenes sintéticas y propone una solución útil para la detección de manipulaciones faciales.

En la figura 4 podemos observar el Pipeline del proyecto.



Fig. 4: Pipeline FaceForensics++

El estado del arte en la detección de imágenes modificadas mediante el uso de técnicas de aprendizaje profundo se centra en el desarrollo y mejora de los modelos existentes,

la exploración de nuevas técnicas de aprendizaje profundo, el uso de grandes conjuntos de datos de entrenamiento y la investigación en detección de manipulación específica.

5 DATASET

La elección de un conjunto de datos adecuado es una parte crítica en la creación de un modelo. La calidad y la representatividad de los datos de entrenamiento son factores clave en la precisión y la eficacia del modelo.

Este punto del proyecto es bastante delicado, ya que este influirá de una forma considerable a nuestro futuro modelo, por lo tanto debemos de tener en cuenta lo siguiente.

- 1. El conjunto de datos determina la capacidad de generalización del modelo:** Si el conjunto de datos de entrenamiento es demasiado pequeño o no es representativo de las diferentes posibilidades de modificación de imágenes, el modelo puede tener dificultades para detectar imágenes modificadas en el mundo real. Un conjunto de datos amplio y variado puede mejorar la capacidad de generalización del modelo y permitir que se ajuste mejor a diferentes situaciones.
- 2. El conjunto de datos influye en la precisión:** Si el conjunto de datos de entrenamiento contiene imágenes mal etiquetadas o ruidosas, el modelo puede ser menos preciso en la detección de imágenes modificadas. Por lo tanto, es importante elegir un conjunto de datos de alta calidad y con etiquetas precisas.
- 3. El conjunto de datos influye en el rendimiento:** Un conjunto de datos grande y variado puede mejorar el rendimiento del modelo, permitiendo una mejor detección de imágenes modificadas y reduciendo el riesgo de sobreajuste.
- 4. El conjunto de datos puede influir en el tipo de modelo que se utiliza:** El conjunto de datos puede influir en la elección del tipo de modelo que se utiliza para la detección de imágenes modificadas. Por ejemplo, si el conjunto de datos contiene una gran cantidad de imágenes con modificaciones sutiles, puede ser más apropiado utilizar un modelo basado en GAN o en aprendizaje por transferencia.

5.1. Dataset elegido

Después de analizar varios datasets, he decidido elegir **Casia-2** [6]. Este conjunto de imágenes es idóneo para crear un modelo robusto, ya que contiene distintas categorías tanto originales como modificadas, y en cada una de ellas se aplican diferentes tipos de modificaciones, como recortes y pegados de la misma imagen (**Same**) o de otra imagen (**Different**).

Las categorías incluidas en el dataset son las siguientes:

- Animales.
- Arquitectura.
- Arte.
- Personas.

- Interiores.
- Plantas.
- Textil.

En la figura 5, se muestra un ejemplo de cada una de estas categorías en el conjunto de originales.



Fig. 5: Ejemplos de cada categoría en el conjunto original.

Para tener una idea del conjunto modificado, se puede observar la figura 6.



Fig. 6: Ejemplos de cada categoría en el conjunto modificado.

Sin embargo, este dataset presenta un problema, ya que está algo desequilibrado en cuanto a las categorías, no te-

nemos la misma cantidad de imágenes para cada categoría. Este problema se puede apreciar en la figura 7.

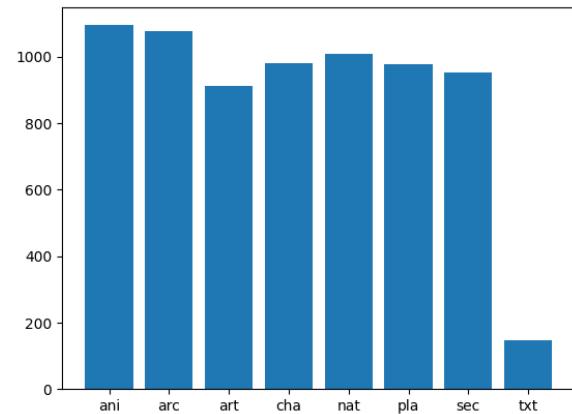


Fig. 7: Cantidad de imágenes por categoría en el conjunto original.

Afortunadamente, esto no representa una gran complicación, ya que se ha decidido recortar el dataset limitando a 600 imágenes por cada categoría. De esta forma, se equilibra el conjunto. Sin embargo, la categoría de **Textil** cuenta con muy pocas imágenes, por lo que se ha optado por eliminarla para simplificar el problema. El resultado se muestra en la figura 8.

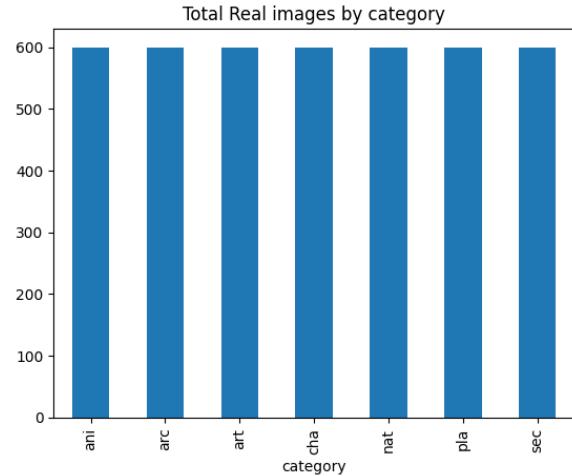


Fig. 8: Cantidad de imágenes por categoría en el conjunto original después de corregirlo.

El mismo problema ocurre en las imágenes de cada categoría en el conjunto modificado, pero en este caso, hay una variable adicional: no todas las categorías tienen la misma cantidad de imágenes con distintas modificaciones. Este desequilibrio se puede apreciar mejor en la figura 9.

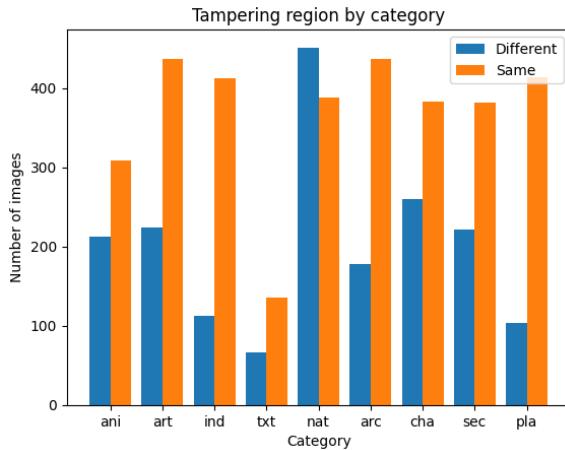


Fig. 9: Cantidad de imágenes por categoría y método del conjunto modificado.

Aplicando la misma lógica que en el set de las originales, recortamos a 600 imágenes por categoría pero además de estas 600 la mitad serán de una modificación u otra.

En el cual tenemos el resultado que muestra la figura 10

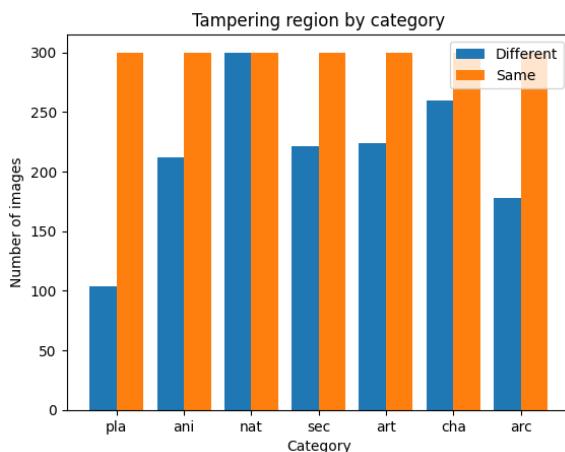


Fig. 10: Cantidad de imágenes por categoría y método del conjunto modificado tras corregirlo.

El resultado no es el ideal, pero es el mejor que se puede obtener con estos datos.

5.2. Alternativas

Lamentablemente, encontrar conjuntos de datos que cumplan con los requisitos necesarios para desarrollar un modelo robusto ha sido un desafío. La mayoría de los conjuntos de datos disponibles eran de pago o demasiado específicos para nuestra investigación. Por lo tanto, se optó por utilizar el conjunto de datos **Casia-2** [6], que cumplió con nuestras necesidades.

Aunque existe una alternativa al conjunto de datos seleccionado, que es la versión anterior, **Casia-1** [7], se decidió no utilizarla. Esto se debe a que los propios creadores del conjunto de datos indican que la calidad del contenido de esta versión es inferior a la de la versión 2.

6 TÉCNICAS

Durante este apartado se explicarán las técnicas usadas durante todo el trabajo y en qué consisten estas.

6.1. Análisis de nivel de error

La técnica de análisis de nivel de error (**Error Level Analysis, ELA**)[8] es una técnica forense que se utiliza para identificar áreas de una imagen que puedan haber sido manipuladas. El principio detrás de esta técnica es que cuando una imagen se comprime y se guarda en un formato con pérdida, como JPEG, la imagen se divide en bloques de píxeles y se aplica un nivel de compresión a cada bloque.

Si una imagen se ha modificado y se ha guardado nuevamente, se espera que el nivel de compresión de las áreas modificadas sea diferente al de las áreas originales. La técnica ELA compara los niveles de compresión en diferentes áreas de la imagen y resalta las áreas donde los niveles de compresión son diferentes, lo que sugiere una posible manipulación.

Podemos observar el resultado que se obtiene al aplicar esta técnica a una imagen modificada en la 11

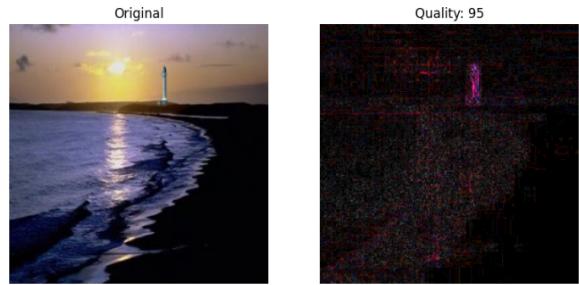


Fig. 11: Aplicando ELA a 95 % a una Imagen modificada.

Como se puede observar en la imagen resultante de aplicar ELA, la modificación realizada en la imagen corresponde a la arquitectura de un faro al fondo de la imagen.

6.2. Wavelet

La técnica de **Wavelet** [9] es una herramienta matemática ampliamente utilizada en el procesamiento de señales y el análisis de imágenes. Su objetivo principal es descomponer una señal o imagen en diferentes componentes de frecuencia, lo que permite un análisis más detallado y eficiente de la información contenida en la imagen.

En el contexto del análisis de imágenes, la transformada de **Wavelet** puede detectar patrones y características relevantes, lo que la hace muy útil en aplicaciones de procesamiento de imágenes y visión por computadora.

Al aplicar la transformada de **Wavelet** a una imagen, se obtienen principalmente tres imágenes que representan diferentes niveles de detalle: **LH (codificación horizontal)**, **HL (codificación vertical)** y **HH (codificación diagonal)**. Cada una de estas imágenes muestra detalles de alta frecuencia en diferentes orientaciones y direcciones, lo que proporciona información valiosa sobre la estructura y

textura de la imagen.

En la figura 12 se muestra un ejemplo de cómo se aplica la transformada de **Wavelet** a una imagen modificada y se obtienen las tres imágenes de detalle correspondientes. Estas imágenes pueden ser utilizadas en diferentes aplicaciones, como la detección de objetos y la eliminación de ruido, entre otras.



Fig. 12: Aplicando Wavelet a una Imagen modificada (Original - LH - HL - HH).

7 RESULTADOS

Antes de analizar en detalle los resultados obtenidos en este trabajo, es importante mencionar que se utilizó una GPU **NVIDIA RTX 3060** y una CPU **AMD Ryzen 5600X**, lo que permitió reducir significativamente el tiempo de entrenamiento y evaluación de los modelos de deep learning empleados.

Se realizaron diversas pruebas utilizando diferentes combinaciones de técnicas de preprocessamiento de imágenes y modelos de clasificación para determinar cuál era la más efectiva para identificar imágenes manipuladas. Tras evaluar los resultados, se encontró que la técnica de preprocessamiento de **Error Level Analysis (ELA)** combinada con el modelo de clasificación base de **MobileNet** [10] tuvo un desempeño significativamente superior al de otras técnicas de preprocessamiento y otros modelos.

Para comprender mejor este modelo, es importante tener en cuenta que está compuesto de aproximadamente **4,279,489** parámetros, de las cuales **4,257,601** son entrenables. En este caso, se ha modificado la entrada original de **224x224x3** a una entrada de **256x256x3**, lo que permite al modelo tener más información de píxeles para mejorar su precisión sin sacrificar demasiado en términos de eficiencia.

Es relevante mencionar que este modelo utiliza **transfer learning** [11] del modelo **MobileNet**, una técnica que permite aprovechar los conocimientos previos de una red neuronal ya entrenada y ajustarla a una tarea específica.

En la figura 13 se muestra un resumen del modelo creado, lo que proporciona una visión general de su arquitectura y parámetros clave.

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[None, 256, 256, 3]	0
conv1 (Conv2D)	(None, 128, 128, 32)	864
conv1_bn (BatchNormalizatio n)	(None, 128, 128, 32)	128
conv1_relu (ReLU)	(None, 128, 128, 32)	0
conv_dw_1 (DepthwiseConv2D)	(None, 128, 128, 32)	288
conv_dw_1_bn (BatchNormaliz ation)	(None, 128, 128, 32)	128
conv_dw_1_relu (ReLU)	(None, 128, 128, 32)	0
conv_pw_1 (Conv2D)	(None, 128, 128, 64)	2048
conv_pw_1_bn (BatchNormaliz ation)	(None, 128, 128, 64)	256
...		
Total params:	4,279,489	
Trainable params:	4,257,601	
Non-trainable params:	21,888	

Fig. 13: Summary del Modelo.

Los resultados finales de la evaluación se resumen en la tabla 1, donde se puede apreciar que la técnica **ELA** logró una tasa superior respecto a la segunda técnica **Wavelet**. Este resultado indica que la técnica **ELA** es especialmente efectiva para identificar imágenes manipuladas, debido a que resalta las posibles alteraciones en la imagen mejor que otras técnicas de preprocessamiento.

Modelo	Preprocesamiento	Épocas	Accuracy
MobileNet	ELA	9	0.89
VGG16	ELA	15	0.87
ResNet50	ELA	7	0.83
Scratch	ELA	12	0.74
Scratch	Wavelet	13	0.62

Table 1: RESULTADOS

En la tabla 2 se presentan los tiempos de entrenamiento para los diferentes modelos. Cabe destacar que estos tiempos no son muy elevados debido a la utilización de una técnica llamada **EarlyStop** [12], la cual evita el **overfitting** [13] del modelo, permitiendo que éste pueda generalizar en lugar de memorizar.

Modelo	Preprocesamiento	Tiempo de entrenamiento
MobileNet	ELA	4 min 38s
VGG16	ELA	36 min 11s
ResNet50	ELA	10 min 46s
Scratch	ELA	8 min 31s
Scratch	Wavelet	2 min 5s

Table 2: TIEMPOS DE ENTRENAMIENTO

Gracias a la técnica de **EarlyStop**, los modelos han sido capaces de generalizar y obtener buenos resultados en un tiempo de entrenamiento razonable.

Si observamos la figura 14 podemos observar como usando un preprocessamiento **ELA** junto a la arquitectura que nos proporciona el modelo de **MobileNet** se han obtenido las siguientes métricas.

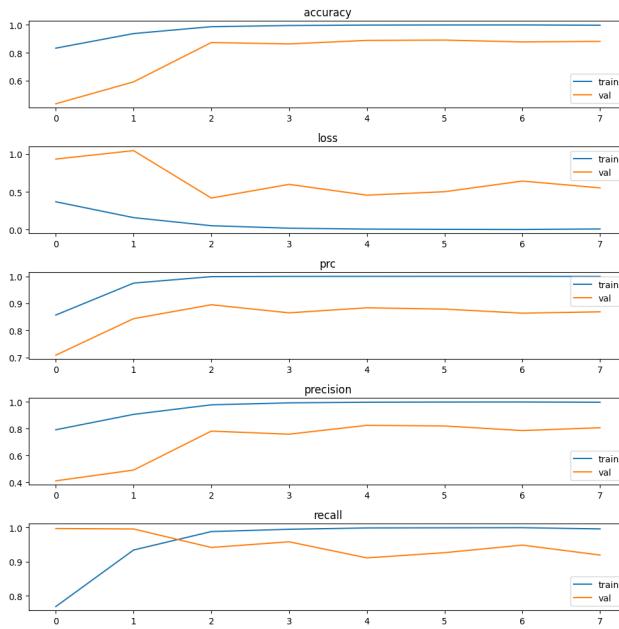


Fig. 14: metricas MobileNet con preprocesamiento ELA

Para una mejor compresión de las métricas podemos observar en la figura 15 en la cual nos muestra la matriz de confusión.

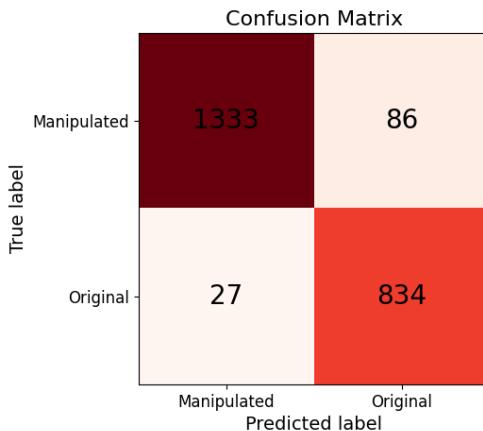


Fig. 15: Matriz de confusión MobileNet con preprocesamiento ELA

La matriz de confusión muestra que el modelo ha sido capaz de clasificar con éxito 2.167 imágenes de un total de 2.280, lo que equivale aproximadamente al **99,95 %**. También se puede ver que el modelo se ha equivocado en tan solo 113 imágenes, lo que representa el **0,05 %** del total de las imágenes clasificadas.

Como trabajo extra se ha desarrollado una herramienta de IA que puede ser utilizada como apoyo en el análisis forense de imágenes modificadas. Los usuarios finales pueden utilizar esta herramienta para detectar posibles modificaciones en las imágenes a través de un mapa de calor, en el que la parte más roja indica las zonas donde el modelo identifica posibles alteraciones.

En el cual este tiene un rendimiento bastante elevado ya

que el tiempo de procesamiento y evaluación de la imagen RGB con unas dimensiones de 256x256 pixeles es de **7 segundos**.

Para ilustrar mejor esta idea, se muestra en la figura 16 una imagen modificada junto con el mapa de calor mencionado anteriormente.

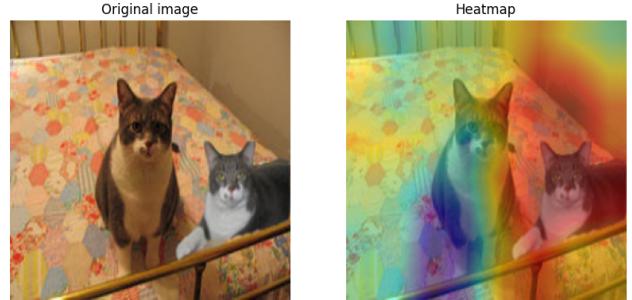


Fig. 16: Mapa de calor MobileNet con preprocesamiento ELA

Dado a la misma naturaleza de la imagen se puede deducir que la modificación se encuentra en uno de los dos gatos apreciados en la imagen, más concretamente el de más a la derecha, se puede ver como el modelo aproximadamente marca con una zona roja ese sector, esto es una primera aproximación a lo que puede ser una herramienta de IA en la cual puede ayudarnos a poder detectar mejor las modificaciones o indicar por donde deberíamos de realizar un análisis más preciso.

Aunque los resultados obtenidos con el modelo mencionado anteriormente hayan sido los más destacados, es importante destacar que se llevaron a cabo diversas pruebas utilizando diferentes técnicas de preprocesamiento y otros modelos para detectar modificaciones.

Para no extender demasiado este informe, se presentarán los resultados de estos diferentes modelos junto con una explicación breve de cada uno. Esto permitirá tener una visión más amplia del desempeño de los diferentes enfoques, lo que puede ayudar a identificar áreas de mejora y futuras investigaciones.

Empezaremos con el modelo **VGG16** [14] con un preprocesamiento de **ELA**, si observamos la tabla 3 veremos sus distintas métricas.

Métrica	Resultado
Accuracy	0.87
Loss	0.38
PRC	0.87
Precision	0.80
Recall	0.86

Table 3: METRICAS VGG16.

Como se puede observar en la figura 3, los resultados del modelo VGG16 son bastante similares a los de MobileNet, aunque ligeramente inferiores.

El siguiente modelo que se analizó fue ResNet50 [15], el cual demostró ser ligeramente menos efectivo que nuestro mejor modelo.

En la tabla 4, se pueden ver las métricas correspondientes que evidencian este resultado.

Metrica	Resultado
Accuracy	0.83
Loss	0.46
PRC	0.87
Precision	0.81
Recall	0.72

Table 4: METRICAS RESNET50.

Para finalizar este análisis, se presentarán las métricas del modelo creado desde cero, también conocido como "Scratch", el cual no utiliza Transfer Learning. Es importante tener en cuenta que este modelo no cuenta con un dataset tan grande como el de MobileNet o VGG16, por lo que es probable que sus resultados sean inferiores.

En la tabla 5, se muestran las métricas obtenidas de este modelo.

Metrica	Resultado
Accuracy	0.74
Loss	1.64
PRC	0.82
Precision	0.94
Recall	0.32

Table 5: METRICAS SCRATCH.

Para entender mejor porque este resultado a continuacion se muestra en la figura 17 la arquitectura del modelo **Scratch**.

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 252, 252, 32)	2432
batch_normalization (BatchN ormalization)	(None, 252, 252, 32)	128
activation (Activation)	(None, 252, 252, 32)	0
max_pooling2d (MaxPooling2D)	(None, 126, 126, 32)	0
conv2d_1 (Conv2D)	(None, 122, 122, 64)	51264
batch_normalization_1 (Bathc hNormalization)	(None, 122, 122, 64)	256
activation_1 (Activation)	(None, 122, 122, 64)	0
max_pooling2d_1 (MaxPooling 2D)	(None, 61, 61, 64)	0
conv2d_2 (Conv2D)	(None, 57, 57, 128)	204928
...		
Total params:	25,951,169	
Trainable params:	25,950,209	
Non-trainable params:	960	

Fig. 17: Summary del Modelo Scratch.

8 CONCLUSIONES

REFERENCIAS

- [1] Página web de Adobe Analytics. <https://business.adobe.com/uk/products/analytics/adobe-analytics.html>
- [2] Noticia relevante del medio de Xataka sobre el software que usa adobe para la detección de imágenes modificadas mediante Photoshop. <https://acortar.link/PeLl6m>
- [3] Paper de detección de imágenes modificadas mediante Photoshop. <https://acortar.link/e77cev>
- [4] Paper explicativo de como funciona la técnica de Deep Fake. <https://arxiv.org/pdf/1909.11573.pdf>
- [5] Paper donde proponen una resolución a las imágenes modificadas apoyándose en DeepFakes, Face2Face, FaceSwap y NeuralTextures. <https://arxiv.org/abs/1901.08971>
- [6] Repositorio de Github donde se ha obtenido el Dataset. <https://github.com/namtpham/casia2groundtruth>
- [7] Repositorio de una alternativa al Dataset escogido. <https://github.com/namtpham/casialgroundtruth>
- [8] Explicación mas detalladamente de la técnica ELA https://forensics.map-base.info/report_2/index_en.shtml
- [9] Explicación mas detalladamente e implementación de la técnica Wavelet <https://acortar.link/cq8jPp>
- [10] Explicación mas detallada de como funciona el modelo de MobileNet <https://acortar.link/rqd8zc>
- [11] Blog donde se explica que es y como funciona el transfer learning <https://www.tokioschool.com/noticias/transfer-learning/>
- [12] Explicación mas detallada de que es el earlyStop en deep learning y como implementarlo <https://acortar.link/w8QGLe>
- [13] Paper donde detalla las perjudicaciones de tener overfitting en un modelo y como poder solucionarlo. <https://iopscience.iop.org/article/10.1088/1742-6596/1168/2/022022/pdf>
- [14] Explicación mas detallada de como funciona el modelo de VGG16. <https://acortar.link/t85jom>
- [15] Explicación mas detallada de como funciona el modelo de ResNet50. <https://datagen.tech/guides/computer-vision/resnet-50/>

APÈNDIX

A.1. Secció d'Apèndix

.....
.....
.....
.....

A.2. Secció d'Apèndix

.....
.....
.....
.....