

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE SISTEMAS

UNIDAD DE TITULACIÓN

**MODELO DE CLASIFICACIÓN SUPERVISADO DE FOTOGRAFÍAS
DE FACHADAS PARA EVALUAR EL DAÑO ESTRUCTURAL
OCASIONADO POR SISMOS DE ACUERDO CON LA ESCALA
MACROSÍSMICA EUROPEA PARA APOYO DE TOMA DE
DECISIONES EN EL INSTITUTO GEOFÍSICO-EPN**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DE GRADO DE
MAGISTER EN SISTEMAS DE INFORMACIÓN MENCION INTELIGENCIA DE
NEGOCIOS Y ANALÍTICA DE DATOS MASIVOS**

BLANCA DAMARIS TARAPUES FUEL

blanca.tarapues@epn.edu.ec

Director: Dra. Lorena Recalde

lorena.recalde@epn.edu.ec

Codirector: Dr. Edison Loza

edison.loza@epn.edu.ec

2021

APROBACIÓN DEL DIRECTOR

Como directora del trabajo de titulación Modelo de clasificación supervisado de fotografías de fachadas para evaluar el daño estructural ocasionado por sismos de acuerdo con la Escala Macrosísmica Europea para apoyo de toma de decisiones en el Instituto Geofísico - EPN desarrollado por Blanca Damaris Tarapues Fuel, estudiante de la Maestría en Sistemas de Información Mención en Inteligencia de Negocios y Analítica de Datos Masivos, habiendo supervisado la realización de este trabajo y realizado las correcciones correspondientes, doy por aprobada la redacción final del documento escrito para que prosiga con los trámites correspondientes a la sustentación de la Defensa Oral.



Dra. Lorena Recalde
DIRECTORA

APROBACIÓN DEL CODIRECTOR

Como codirector del trabajo de titulación Modelo de clasificación supervisado de fotografías de fachadas para evaluar el daño estructural ocasionado por sismos de acuerdo con la Escala Macrosísmica Europea para apoyo de toma de decisiones en el Instituto Geofísico - EPN desarrollado por Blanca Damaris Tarapues Fuel, estudiante de la Maestría en Sistemas de Información Mención en Inteligencia de Negocios y Analítica de Datos Masivos, habiendo supervisado la realización de este trabajo y realizado las correcciones correspondientes, doy por aprobada la redacción final del documento escrito para que prosiga con los trámites correspondientes a la sustentación de la Defensa Oral.



Dr. Edison Loza
CO-DIRECTOR

DECLARACIÓN DE AUTORÍA

Yo, Blanca Damaris Tarapues Fuel, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Escuela Politécnica Nacional puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.



Blanca Damaris Tarapues Fuel

DEDICATORIA

A mi esposo, Diego, y a mis futuros hijos.

AGRADECIMIENTO

Agradezco a Dios, por su infinita misericordia y bondad.

A mi esposo, por su apoyo y amor.

A mis padres, por heredarme la valentía y deseo de seguir luchando siempre.

A mis directores de tesis, Dra. Lorena Recalde y Dr. Edison loza, no sólo por su impecable labor de dirección sino también por su continua enseñanza y apoyo en la elaboración de este trabajo.

ÍNDICE DE CONTENIDO

1. INTRODUCCIÓN	1
1.1. OBJETIVO GENERAL.....	1
1.2. OBJETIVOS ESPECÍFICOS.....	1
1.3. ALCANCE.....	1
1.4. MARCO TEÓRICO.....	1
1.4.1. DAÑO SÍSMICO.....	1
1.4.2. APRENDIZAJE DE MÁQUINA Y APRENDIZAJE PROFUNDO.....	2
1.4.3. REDES NEURONALES.....	5
1.4.4. REDES NEURONALES CONVOLUCIONALES.....	6
1.4.5. PROCESAMIENTO DE IMÁGENES.....	7
1.4.5.1. RUIDO.....	7
1.4.5.2. FILTRO.....	8
1.4.5.3. MODELOS DE COLOR.....	9
1.4.6. TRABAJOS RELACIONADOS.....	9
2. METODOLOGÍA	11
2.1 ANÁLISIS Y COMPRENSIÓN DEL NEGOCIO.....	12
2.2 COMPRENSIÓN DE LOS DATOS.....	13
2.2.1 RECOLECCIÓN DEL CONJUNTO DE DATOS.....	13
2.2.1.1 DESCRIPCIÓN DEL CONJUNTO DE DATOS.....	15
2.2.1.2 EXPLORACIÓN DEL CONJUNTO DE DATOS.....	15
2.3 PREPARACIÓN DE LOS DATOS.....	15
2.3.1 SELECCIÓN DE DATOS.....	15
2.3.2 CONSTRUCCIÓN DEL CONJUNTO DE DATOS.....	18
2.3.2.1 FORMATO DE IMÁGENES.....	18
2.3.2.2 TAMAÑO DE IMÁGENES.....	19
2.3.2.3 DIVISIÓN DEL CONJUNTO DE DATOS.....	19
2.4 PREPROCESAMIENTO DE LOS DATOS.....	19
2.4.1 CONVERSIÓN DE UNA IMAGEN A COLOR RGB.....	19
2.4.2 SUAVIZADO DE LA IMAGEN.....	20
2.4.3 AUMENTO DE DATOS.....	21

2.5	MODELADO	21
2.5.1	SELECCIÓN DE HERRAMIENTAS	21
2.5.2	ARQUITECTURA DE LA RED NEURONAL CONVOLUCIONAL	22
2.5.2.1	APRENDIZAJE POR TRANSFERENCIA	22
2.5.2.2	ARQUITECTURA USADA	25
2.5.2.3	ENTRENAMIENTO Y OPTIMIZACIÓN DE LA RED NEURONAL CONVOLUCIONAL.....	26
2.6	EVALUACIÓN DEL MODELO.....	27
3.	RESULTADOS Y DISCUSIÓN	28
3.1.	RESULTADOS DE ENTRENAMIENTO CON APRENDIZAJE POR TRANSFERENCIA (SIN AUMENTO DE DATOS).....	28
3.1.1.	PRECISIÓN EQUILIBRADA EN EL CONJUNTO DE DATOS MA Y HA.....	31
3.1.2.	F1-SCORE EN EL CONJUNTO DE DATOS MA Y HA	32
3.2.	RESULTADOS DE ENTRENAMIENTO CON APRENDIZAJE POR TRANSFERENCIA (CON AUMENTO DE DATOS).....	36
3.2.1.	PRECISIÓN EQUILIBRADA CON AUMENTO DE DATOS MA ₁ Y HA ₁	38
3.2.2.	F1-SCORE CON AUMENTO DE DATOS MA ₁ Y HA ₁	39
3.2.3.	F1-SCORE CON AUMENTO DE DATOS CLASE POR CLASE MA ₁ Y HA ₁	40
3.3.	DISCUSIÓN	44
4.	CONCLUSIONES Y RECOMENDACIONES	46
4.1.	CONCLUSIONES.....	46
4.2.	RECOMENDACIONES.....	47
ANEXOS	54

LISTA DE FIGURAS

Figura 1. Tabla de Vulnerabilidad [5].....	2
Figura 2. Inteligencia Artificial, Aprendizaje de Máquina, Aprendizaje Profundo [9]2	
Figura 3. Neurona Artificial [15].....	5
Figura 4. Anatomía de una Red Neuronal [15].	5
Figura 5. Arquitectura de una red Neuronal Convolutacional (CNN). Adaptado de [16].....	7
Figura 6. Metodología CRISP-DM. Adaptado de [26].....	11
Figura 7. Número de Eventos Sísmicos por Quinquenio y Magnitud ≥ 6.0 [32]. .	13
Figura 8. Terremoto del 10 de diciembre 1970 en Loja [34].....	14
Figura 9. Tamaño de una misma imagen con diferentes tipos de formatos.	18
Figura 11. Filtro Promedio	20
Figura 10. Filtro Gaussiano	20
Figura 12. Accuracy MA.....	30
Figura 13. Accuracy HA	30
Figura 14. Accuracy Balance - MA.....	32
Figura 15. Accuracy Balanced – HA.....	32
Figura 16. F1-Score - MA	33
Figura 17. F1-Score - HA.....	33
Figura 18. Comparación Clase por Clase F1-Score MA	34
Figura 19. Comparación Clase por Clase F1-Score HA.....	34
Figura 20. Comparativa Métricas de Clasificación MA.....	35
Figura 21. Comparativa Métricas de Clasificación HA	35
Figura 22. Accuracy MA ₁	37
Figura 23. Accuracy - HA ₁	37
Figura 24. Accuracy Balanced MA ₁	38
Figura 25. Accuracy Balance HA ₁	38
Figura 26. F1-Score MA ₁	39
Figura 27. F1 - Score HA ₁	39
Figura 28. Comparativa Métricas de Clasificación MA ₁	40

Figura 29. Comparativa Métricas de Clasificación HA1	40
Figura 30. Comparativa Clase por Clase de F1-Score en MA ₁	41
Figura 31. Comparativa Clase por Clase de F1-Score en HA1	41
Figura 32. Comparación Resultados Final Métricas MA vs MA ₁	43
Figura 33.Comparación Resultados Final Métricas HA vs HA ₁	43
Figura 34. Diagrama de Flujo para Construir un Modelo de Aprendizaje Autónomo.	44

LISTA DE TABLAS

Tabla 1. Clasificación de Daños en Edificios de Mampostería	3
Tabla 2. Clasificación de Daños en Edificios de Fábrica [5].....	4
Tabla 3. Intensidades del Sismo Según el Grado de Daño y Tipo de Vulnerabilidad [5].....	5
Tabla 4. Fuente de Datos de Imágenes de Edificios con Daños Estructurales.....	14
Tabla 5. Distribución del Conjunto de Datos según el Grado y Tipo de Estructura	15
Tabla 6. Clasificación Manual de Imágenes.....	17
Tabla 7. Distribución del Conjunto de Datos de Tipo Mampostería según el Formato.	18
Tabla 8. Distribución del Conjunto de Datos de Tipo Hormigón Armado según el Formato	19
Tabla 9. Resultado de entrenamiento, Validación, y Prueba MA	29
Tabla 10. Resultados de Entrenamiento, Validación, y Prueba HA	29
Tabla 11. Promedio de Métricas de Clasificación sobre el Conjunto de Datos MA	31
Tabla 12. Promedio de Métricas de Clasificación sobre el Conjunto de Datos HA31	
Tabla 13. Resultados de entrenamiento, validación y prueba MA ₁	36
Tabla 14. Resultados del entrenamiento, validación y prueba HA ₁	36
Tabla 15. Métricas de Evaluación para el Conjunto de Datos MA y MA ₁	42
Tabla 16. Métricas de Evaluación para el Conjunto de Datos HA y HA ₁	42

LISTA DE ANEXOS

Anexo I–Script compilación del modelo	55
Anexo II – Script Codificación One-hot	55

RESUMEN

La estimación del grado de daño de estructuras de edificios después de un evento sísmico es un requerimiento muy importante dentro del campo de la Sismología, ya que es un parámetro empleado para determinar la intensidad de un sismo.

El objetivo principal del presente trabajo de titulación es diseñar un modelo de clasificación supervisada de fotografías de fachadas que permita evaluar el daño estructural ocasionado por sismos. Para ello se aplica técnicas de Aprendizaje de Máquina y Aprendizaje Profundo.

En primera instancia, cada imagen es preprocesada mediante técnicas de filtrado no lineal para, a partir de ello, obtener un conjunto de imágenes con alta probabilidad de contener información que aporte al modelo. Este preprocesamiento reduce el ruido, y realiza un suavizado de la imagen. Luego, las imágenes son utilizadas para crear un modelo gracias a una red neuronal convolucional, el cual permitirá clasificar cada imagen según el grado de daño, comprendido entre uno a cinco.

De un total de 605 imágenes candidatas, la red clasificó correctamente el 80% de imágenes clasificadas manualmente. Esta precisión alcanzada indica que el modelo desarrollado constituye un paso importante hacia la implementación de un futuro sistema de clasificación de imágenes de edificios destruidos por un sismo.

Palabras clave: grado de daño, aprendizaje profundo, redes neuronales convolucionales, aprendizaje por transferencia.

ABSTRACT

The estimation of the degree of damage of building structures after a seismic event is a very coveted objective within the field of seismology, since it is a parameter used to determine the intensity of an earthquake.

The main objective of this work is to design a supervised classification model of façade photographs to evaluate the structural damage caused by earthquakes. For this purpose, Machine Learning and Deep Learning methods are applied.

In the first instance, each image is preprocessed using nonlinear filtering techniques to obtain a set of candidate images with a high probability of containing information that contributes to the model. The preprocessing reduces noise and performs image smoothing. Then, the images are used to build a model through a convolutional neural network. The model will allow classify images according to the degree of damage, ranging from one to five.

Out of a total of 605 candidate images, the network correctly classified 80% of the manually classified images. The accuracy achieved indicates that the model developed is a good step in the implementation of a future system for classifying images of buildings destroyed by an earthquake.

Keywords: Degree of Damage, Deep Learning, Convolutional Neural Networks, Transfer Learning, Preprocessing.

1. INTRODUCCIÓN

Hoy en día, la tecnología es una herramienta básica para el diario vivir y es parte fundamental de la sociedad. Ésta se apoya en los recursos que la misma ofrece como son los sistemas de información. Estos sistemas, mediante el uso de las tecnologías de la información dan soporte a las operaciones empresariales, la gestión y la toma de decisiones, proporcionando a las organizaciones la información que necesitan [1].

La toma de decisiones es un proceso en el que una o más personas identifican un problema y entonces diseñan, eligen, implementan y programan una solución. Dichos problemas pueden ser provocados por factores internos o externos a la organización, como los desastres naturales [2]. Los desastres naturales son sucesos catastróficos derivados de causas naturales como erupciones volcánicas, tornados, terremotos, etc., sobre los que el hombre no tiene control [3]. Así, cada año, por ejemplo, más de 100.000 terremotos pueden ser sentidos por la gente alrededor del mundo [4]. Estos terremotos van desde eventos muy pequeños que sólo sienten unos pocos individuos hasta grandes movimientos que destruyen ciudades enteras. El número de vidas que se pierden y el importe de las pérdidas económicas que se derivan de un terremoto dependen del tamaño, la profundidad y la ubicación del terremoto, la intensidad de las sacudidas del suelo y los efectos relacionados con el inventario de edificios, y la vulnerabilidad de ese inventario de edificios a los daños. Los efectos más intensos de un terremoto suelen producirse en los lugares más cercanos a la zona de la falla que lo produjo. A medida que aumenta la magnitud del terremoto, también lo hace la intensidad máxima de los efectos producidos y el tamaño del área geográfica que experimenta estos efectos.

Actualmente, existen diferentes escalas que atribuyen al sismo un grado según la gravedad de los daños observados. Entre ellas están: la escala modificada de Mercalli, la escala MSK (Medvédev-Sponheuer-Kárník), y la Escala Macrosísmica Europea (EMS) [2]. Estas escalas son usadas en mayor o menor medida en las diferentes regiones del globo. En Ecuador, el Instituto Geofísico de la Escuela Politécnica Nacional (IG-EPN) aplica la Escala Macrosísmica Europea (EMS) [1].

La escala EMS fue creada por la Asamblea General de la Comisión Sismológica Europea (ESC) en 1998 [5]. Esta escala cuenta con parámetros relacionados a la construcción como son: el grado de daño y el tipo de estructura. El grado de daño se clasifica en: a) daños despreciables a ligeros, b) daños moderados, c) daños importantes a graves, d) daños muy graves y e) grado de destrucción total (Tabla 1 y 2). El tipo de estructura se clasifica desde la A a la F según el nivel de resistencia o vulnerabilidad de la edificación. Así, una construcción típica de adobe pertenece a la vulnerabilidad A, construcciones de ladrillo y

hormigón armado son de vulnerabilidad A-C, construcciones con diseño sismo resistentes son de vulnerabilidad D y E, y las estructuras con un nivel alto de diseño sismo resistente, es decir con la más alta resistencia a terremotos, pertenecen a la vulnerabilidad F [6]. La tabla de vulnerabilidad muestra en la Figura 1.

Tipo de estructura		Clase de vulnerabilidad					
		A	B	C	D	E	F
Fábrica	piedra suelta o canto rodado	○					
	adobe (ladrillos de tierra)	○—					
	mampostería	...○					
	sillería		—○—				
	sin armar, de ladrillos o bloques	...○—					
	sin armar, con forjados de HA		—○—				
	armada o confinada			...○—			
Hormigón Armado (HA)	estructura sin diseño sismorresistente (DSR)	...—○—					
	estructura con nivel medio de DSR		...—○—				
	estructura con nivel alto de DSR			...—○—			
	muros sin DSR	...○—					
	muros con nivel medio de DSR		...○—				
	muros con nivel alto de DSR			...○—			
Acero	estructuras de acero			...—○—			
Madera	estructuras de madera		...—○—				









 rango probable  clase de vulnerabilidad más probable
 rango de casos menos probables, excepcionales

Figura 1. Tabla de Vulnerabilidad [5]

Para determinar el grado de daño tras la ocurrencia de terremotos es necesario realizar un informe de campo detallado, tomando todas las fotografías posibles que permitan documentar y analizar el comportamiento y la vulnerabilidad de los edificios [6].



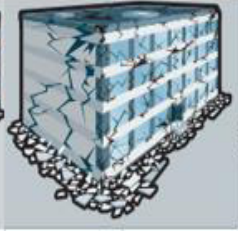

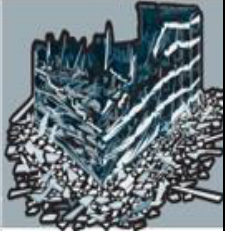
La identificación de la intensidad del sismo, dentro de la EMS, depende hoy por hoy del experto, quien determinará la intensidad del movimiento sísmico en concordancia con la Tabla 3; dejando a criterio del experto lo que significan términos como “pocos” o “muchos” que se incluyen en la evaluación.

Tabla 1. Clasificación de Daños en Edificios de Mampostería

				
GRADO 1	GRADO 2	GRADO 3	GRADO 4	GRADO 5
Daño de despreciables a ligeros	Daños moderados	Daños de importantes a graves	Daños muy graves	Destrucción
Ningún daño estructural, daños no-estructurales ligero	Daños estructurales ligeros, daños no-estructurales moderados	Daños estructurales moderados, daños no-estructurales graves	Daños estructurales graves, daños no-estructurales muy graves	Daños estructurales muy graves
<ul style="list-style-type: none"> *Fisuras en muy pocos muros. *Caída sólo de pequeños trozos de revestimiento. *Caída de piedras sueltas de las partes altas de los edificios en muy pocos casos. 	<ul style="list-style-type: none"> *Grietas en muchos muros. *Caída de trozos bastante grandes de revestimiento. *Colapso parcial de chimeneas. 	<ul style="list-style-type: none"> *Grietas grandes y generalizadas en la mayoría de los muros. *Se sueltan tejas del tejado. *Rotura de chimeneas por la línea del tejado. *Se dañan elementos individuales no estructurales (tabiques, hastiales y tejados). 	<ul style="list-style-type: none"> *Se dañan seriamente los muros. *Se dañan parcialmente los tejados y forjados. 	Colapso total o casi total.

En la actualidad, para determinar la intensidad utilizando la EMS, el IG-EPN realiza una evaluación “*in situ*” de los daños causados por el terremoto. Para ello, debe desplazar a todo su equipo de especialistas al lugar de los hechos, lo cual lleva tiempo y consume recursos [7]. Como parte de la recopilación de información *in situ*, se toman fotografías de las construcciones afectadas para tener una idea general de los efectos causados por el terremoto y cuantificar las estructuras agrupándolas por tipo y grado de daño, y así poder finalmente establecer la intensidad del movimiento. En este sentido, en el presente proyecto de titulación se presenta un modelo de clasificación de fotografías de fachada que permite clasificar el grado de daño, ayudando así al experto a determinar la intensidad del sismo de manera más rápida y precisa. El modelo será entrenado a partir de conjuntos de datos etiquetados correspondientes a terremotos anteriores.

Tabla 2. Clasificación de Daños en Edificios de Fábrica [5].

				
GRADO 1	GRADO 2	GRADO 3	GRADO 4	GRADO 5
Daños de despreciables a ligeros	Daños moderados	Daños de importantes a graves	Daños muy graves	Destrucción
Ningún daño estructural, daños no-estructurales ligero	Daños estructurales ligeros, daños no-estructurales moderados	Daños estructurales moderados, daños no-estructurales graves	Daños estructurales graves, daños no-estructurales muy graves	Daños estructurales muy graves
*Fisuras en el revestimiento de pórticos o en la base de los muros. *Fisuras en tabiques y particiones.	*Grietas en vigas y pilares de pórticos y en muros estructurales. * Grietas en tabiques y particiones; caídas de enlucidos y revestimientos frágiles. *Caída de mortero de las juntas de paneles prefabricados.	*Grietas en pilares y en juntas viga/pilar en la base de los pórticos y en las juntas de los muros acoplados. *Desprendimiento de revocos de hormigón, pandeo de la armadura de refuerzo. *Grandes grietas en tabiques y particiones; se dañan paneles de particiones aislados.	*Grandes grietas en elementos estructurales con daños en el hormigón por compresión y rotura de armaduras; fallos en la trabazón de la armadura de las vigas; ladeo de pilares. *Colapso de algunos pilares o de una planta alta.	Colapso de la planta baja o de partes (por ejemplo, alas) del edificio.

Cabe recalcar que, además del modelo, este proyecto contribuye con la recopilación y etiquetado de forma manual de las imágenes que se utilizan para entrenar y evaluar el modelo. Una futura aplicación de este proyecto reside en poder ayudar a clasificar fotos que la ciudadanía puede compartir en las redes sociales acerca de la destrucción a estructuras, después de un terremoto. Esto permitirá por un lado reducir el tiempo para la valoración de la intensidad del sismo, y ahorrar en los recursos necesarios para realizar esta valoración.

Tabla 3. Intensidades del Sismo Según el Grado de Daño y Tipo de Vulnerabilidad [5].

Intensidad	Daños a edificios
I - IV	Sin daños.
V	Daño de grado 1 en pocos edificios de tipo A y/o B.
VI	Muchos edificios de tipo A sufren grado 2. Muchos edificios de tipo B sufren daño de grado 1; pocos de grado 2. Pocos edificios de tipo C sufren de daño de grado 1.
VII	Muchos edificios de tipo A sufren daño de grado 3; pocos de grado 4. Muchos edificios de tipo B sufren daño de grado 2; pocos de grado 3. Muchos edificios de tipo C sufren daño de grado 1; pocos de grado 2. Pocos edificios de tipo D sufren daño de grado 1
VIII	Muchos edificios de tipo A sufren daño de grado 4; pocos de grado 5. Muchos edificios de tipo B sufren daño de grado 3; pocos de grado 4. Muchos edificios de tipo C sufren daño de grado 2; pocos de grado 3. Pocos edificios de tipo D sufren daño de grado 1; pocos de grado 2
IX	Muchos edificios de tipo A sufren daño de grado 5. Muchos edificios de tipo B sufren daño de grado 4; pocos de grado 5. Muchos edificios de tipo C sufren daño de grado 3; pocos de grado 4. Muchos edificios de tipo D sufren daño de grado 2; pocos de grado 3. Muchos edificios de tipo E sufren daño de grado 1; pocos de grado 2.
X	Todos de los edificios de tipo A y B sufren daño de grado 5. Muchos edificios de tipo C sufren daño de grado 4; pocos de grado 5. Muchos edificios de tipo D sufren daño de grado 3; pocos de grado 4. Muchos edificios de tipo E sufren daño de grado 2; pocos de grado 3. Pocos edificios de tipo F sufren daño de grado 1.
XI	Todos los edificios de tipo A están destruidos. Muchos edificios de tipo B sostienen daño de grado 5. Muchos edificios de tipo C sufren daño de grado 4; pocos de grado 5. Muchos edificios de tipo D sufren daño de grado 4; pocos de grado 5. Muchos edificios de tipo E sufren daño de grado 3; pocos de grado 4. Muchos edificios de tipo F sufren daño de grado 2; pocos de grado 3.
XII	Todos los edificios de tipo A y B están destruidos. La mayoría de los de tipo C y D alcanzan grado 5. La mayoría de los de tipo E alcanza grado 4. La mayoría de los de tipo F alcanza grado 3.

1.1. Objetivo general

Diseñar un modelo de clasificación supervisada de fotografías de fachadas para evaluar el daño estructural ocasionado por sismos de acuerdo a la Escala Macrosísmica Europea para apoyo de toma de decisiones en el IG-EPN, aplicando técnicas de Aprendizaje de Máquina y Aprendizaje Profundo.

1.2. Objetivos específicos

- Realizar una revisión de literatura sobre trabajos enfocados en la clasificación de fotografías para evaluación de daño estructural.
- Recolectar un conjunto de fotografías de fachadas de edificaciones con daños estructurales y etiquetarlas de acuerdo con el tipo de estructura y grado de daño.

1.3. Alcance

El alcance del este proyecto implica el uso de algoritmos de clasificación supervisada, de los existentes, este trabajo se centra en la utilización de redes neuronales para la clasificación de imágenes de fachadas de edificios dañados por sismos. Además, éste proyecto no incluye mecanismos de recolección automática de imágenes desde las redes sociales y tampoco el desarrollo de un sistema que permita la interacción con el modelo.

1.4. Marco Teórico

En esta sección se detallan las definiciones expuestas por varios autores relacionados a las diferentes temáticas que se abordan en este proyecto, tales como: Grado Sísmico, Aprendizaje de Máquina, Redes Neuronales, Procesamiento de Imágenes.

1.4.1. Daño Sísmico

El daño se entiende como el grado de degradación o destrucción causado por un fenómeno peligroso sobre las personas, los bienes, los servicios y los sistemas naturales o sociales. Desde el punto de vista estructural, se relaciona con deformaciones irreversibles [8]. El grado de daño ocasionado por un sismo y el tipo de estructura de los edificios afectados son parámetros, entre otros, que se emplean en la EMS para determinar la intensidad del sismo [6]. Por otra parte, el daño que sufre un edificio, producto de un temblor, está condicionado a la necesidad de describir grados de daños que puedan distinguirse fácilmente por el usuario u observador. En la Tabla 1 y 2 se muestran los diferentes grados

atribuibles a estructuras de mampostería y de hormigón armado, las cuales son las más comunes en ciudades. La descripción bajo cada uno de los grados se considera como “fotos en palabras” de los efectos que se esperan para cada nivel de intensidad. Cada efecto descrito en la escala puede considerarse como un diagnóstico, con el cual se pueden comparar los datos. Así, establecer el grado de daño consiste en comparar los datos con las descripciones ideales de la escala y decidir cuál concuerda mejor.

1.4.2. Aprendizaje de Máquina y Aprendizaje Profundo

La Inteligencia Artificial (IA) es un campo general que abarca el aprendizaje automático y el aprendizaje profundo. Aunque la IA ha demostrado ser adecuada para resolver problemas lógicos bien definidos, como jugar al ajedrez, ha resultado ser poco adecuada a la hora de averiguar reglas explícitas para resolver problemas más complejos y difusos, como la clasificación de imágenes, el reconocimiento del habla y la traducción de idiomas. Es por ésto que se desarrolló un nuevo enfoque para abordar estos problemas más complejos: el aprendizaje profundo como se presenta en la Figura 2.

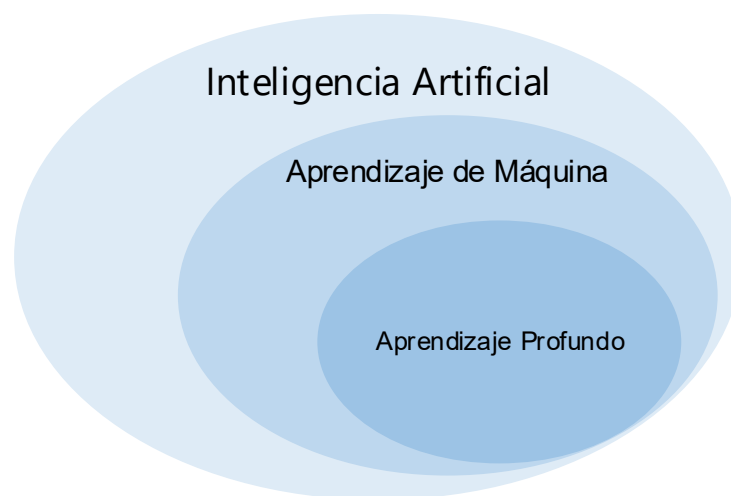


Figura 2. Inteligencia Artificial, Aprendizaje de Máquina, Aprendizaje Profundo [9]

En el Aprendizaje de Máquina un algoritmo se entrena en lugar de programarse explícitamente, es decir, el algoritmo descubre reglas para ejecutar una tarea de procesamiento de datos. Existen diferentes métodos en el Aprendizaje de Máquina que permiten resolver problemas del mundo real, que son: aprendizaje supervisado, aprendizaje no supervisado, aprendizaje semisupervisado, aprendizaje por refuerzo y aprendizaje por transferencia [9].

El aprendizaje supervisado incluye un procedimiento en el que el conjunto de entrenamiento se da como entrada al sistema en el que, cada ejemplo se etiqueta con un valor de salida deseado. Este tipo de aprendizaje se realiza mediante la minimización de una función de pérdida particular, la cual representa el error de salida con respecto al sistema de salida deseado. Una vez completado el entrenamiento, la precisión de cada modelo se mide con respecto a ejemplos disjuntos del conjunto de entrenamiento, también llamado conjunto de validación y/o prueba [10].

El objetivo del aprendizaje no supervisado es que el ordenador aprenda a hacer algo que no le decimos cómo hacer, es decir, consiste en enseñar al agente no dando, categorizaciones explícitas, sino utilizando algún tipo de sistema de recompensa para indicar el éxito. Este tipo de entrenamiento suele encajar en el marco de los problemas de decisión porque el objetivo no es producir una clasificación sino tomar decisiones que maximicen las recompensas [11].

El aprendizaje semisupervisado es un híbrido de los métodos supervisados y no supervisados. Este tipo de aprendizaje se aplica cuando existe gran cantidad de datos pero pocos de ellos están etiquetados, Sin embargo, el aprendizaje semisupervisado aprovecha estos pocos datos etiquetados para crear un modelo inicial que se utiliza para etiquetar la gran cantidad de datos no etiquetados que existen en el dominio. Una vez que se dispone de los datos etiquetados se puede utilizar un algoritmo de aprendizaje automático supervisado para entrenar y crear un modelo final que se utiliza en las tareas de predicción de los datos no visto [9].

El aprendizaje por refuerzo es un método de aprendizaje automático que no es ni aprendizaje supervisado ni aprendizaje no supervisado. Este método consiste en mejorar las estrategias utilizadas para resolver cualquier problema de forma continua basándose en la retroalimentación recibida. Un ejemplo de este método es la máquina que supera al ser humano en un videojuego de Atari. Para lograr esta hazaña, se introdujo un gran número de juegos de ejemplo jugados como entrada al algoritmo que aprendió los pasos a seguir para maximizar la recompensa. La recompensa en este caso es la puntuación final. El algoritmo, tras aprender de las entradas de ejemplo, simula el patrón en cada paso del juego que finalmente maximiza la puntuación obtenida. Este ejemplo es uno de los mayores éxitos del aprendizaje por refuerzo [12].

El aprendizaje por transferencia consiste en reutilizar un modelo construido para realizar una tarea específica que ayuda a resolver otra relacionada [9]. Sin embargo, para lograr mejores medidas de rendimiento, los modelos de Aprendizaje de Máquina deben entrenarse con grandes cantidades de datos etiquetados, ya que la disponibilidad de una menor cantidad de estos supone un menor entrenamiento y el resultado es un modelo con

un rendimiento subóptimo. Por tanto, este método pretende resolver problemas derivados de la disponibilidad de datos mediante la reutilización de los conocimientos obtenidos por otro modelo relacionado. Existen dos formas para lograr este objetivo: reusar un modelo propio o usar un modelo pre-entrenado [9].

Por lo tanto, para realizar el aprendizaje supervisado, se necesitan tres cosas: datos de entrada, ejemplos de resultados esperados, y una forma de medir si el algoritmo está haciendo un buen trabajo. Esta última, es la medición que se utiliza como una señal de retroalimentación para ajustar el funcionamiento del algoritmo. A dicho paso se le conoce como aprendizaje. Ahora bien, los modelos de aprendizaje automático consisten en encontrar representaciones adecuadas para sus datos de entrada, es decir, transformaciones de los datos que los hacen más aptos para la tarea en cuestión, por ejemplo, una tarea de clasificación [13].

Por otro lado, el aprendizaje profundo es un subcampo específico del aprendizaje automático. El aprendizaje profundo se considera como una nueva forma de aprender representaciones a partir de datos que hace hincapié en el aprendizaje de capas sucesivas de representaciones cada vez más significativas. El término "profundo" no hace referencia a ningún tipo de comprensión más profunda lograda por el enfoque; más bien, representa esta idea de capas sucesivas de representaciones. Estas representaciones en capas se aprenden (casi siempre) mediante modelos denominados redes neuronales, estructurados en capas literales apiladas unas sobre otras. De hecho, esta es la razón principal por la que el aprendizaje profundo despegó tan rápidamente, ya que las anteriores técnicas de aprendizaje automático solo implicaban transformación de los datos de entrada en uno o dos espacios, normalmente a través de transformaciones simples como las proyecciones no lineales de alta dimensión, en inglés conocido como Support Vector Machine, SVM o los árboles de decisión [13].

1.4.3. Redes Neuronales

Es un conjunto de elementos de procesamiento, nodos o neuronas como se presenta en la Figura 3, interconectados entre sí por conexiones con un valor numérico y alterable denominado peso, se basa fundamentalmente en sucesivas capas de transformación de datos [14].

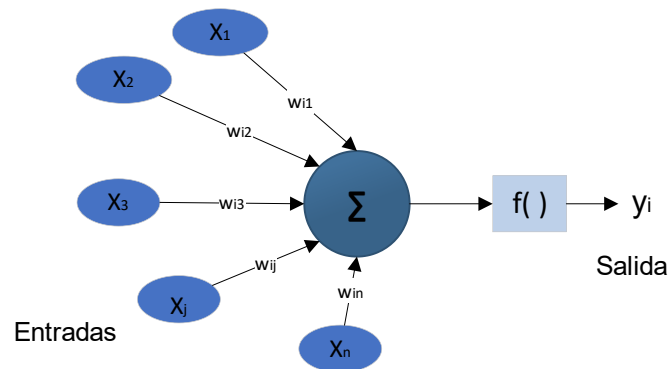


Figura 3. Neurona Artificial [15].

La anatomía de una red neuronal mostrada en la Figura 4 gira en torno a los siguientes objetos:

- Las capas, que se combinan en una red (o modelo).
- Los datos de entrada y los objetivos correspondientes.
- La función de pérdida, que define la señal de retroalimentación utilizada para el aprendizaje.
- El optimizador, que determina cómo procede el aprendizaje.

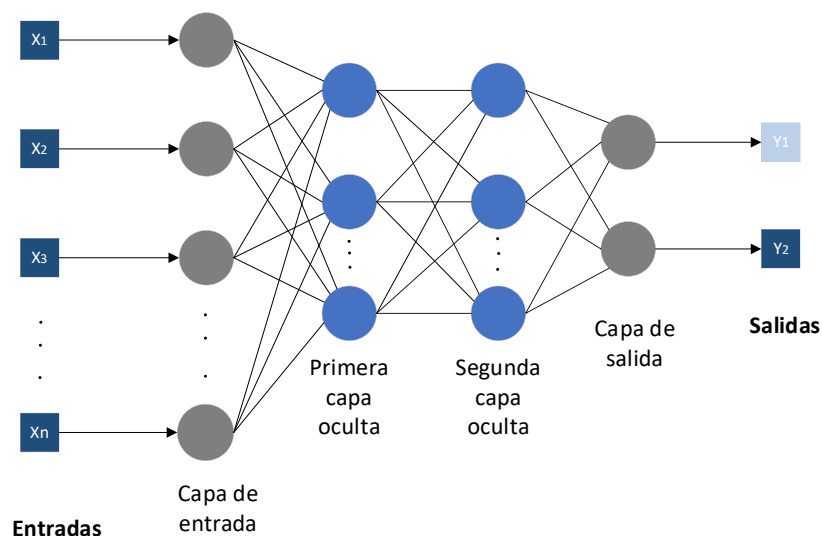


Figura 4. Anatomía de una Red Neuronal [15].

De ese modo, las capas son la estructura de datos fundamental en las redes neuronales, debido a que permite el procesamiento de datos, para ello toma como entrada uno o más tensores y produce, a su vez, uno o más tensores. Es frecuente que estas tengan un estado, el cual consiste en los pesos de la capa y uno o varios tensores. Estos dos juntos contienen el conocimiento de la red. Además, existen diferentes capas para diferentes tipos de procesamiento de datos. Por ejemplo:

- Los datos vectoriales simples, estos se almacenan en tensores de forma 2D, es decir, contienen muestras y características y son procesadas por capas densamente conectadas.
- Los datos secuenciales, estos se almacenan en tensores de forma 3D, es decir, contienen muestras, pasos de tiempo, características y son procesadas por capas recurrentes.
- Los datos de imágenes, estos se almacenan en tensores de forma 4D, y son procesadas por capas de convolución 2D (Conv2D).

Así mismo, el modelo de aprendizaje profundo consiste en definir un espacio de posibilidades, para ello se debe elegir una tipología de red que ayude a restringir el espacio de posibilidades a una serie específica de operaciones tensoriales, mapeando datos de entrada a datos de salida. Las funciones de pérdida representan la cantidad que se minimizará durante el entrenamiento, y constituye una medida de éxito para la tarea en cuestión. El optimizador determina como se actualizará la red basándose en la función de pérdida. Estas dos últimas son la clave para configurar el proceso de aprendizaje [13].

1.4.4. Redes Neuronales Convolucionales

Las redes neuronales convolucionales, en inglés convolucional neuronal network CNN), es un tipo de modelo de aprendizaje profundo para el procesamiento de datos que tienen un patrón cuadrulado, como las imágenes. La CNN es una construcción matemática que suele estar compuesta por tres tipos de capas, convolución, pooling y capas totalmente conectadas. Las dos primeras capas, las capas de convolución y de agrupación, realizan la extracción de características, mientras que la tercera, una capa totalmente conectada, convierte las características extraídas en un resultado final, como la clasificación [16]. La capa de convolución desempeña un papel importante en la CNN, ya que se compone de una pila de operaciones matemáticas, como la convolución. Una convolución es un tipo especializado de operación lineal. Por ejemplo, en las imágenes digitales, los valores de los píxeles se pueden almacenar en una cuadrícula bidimensional (2D), es decir, en una matriz de números 2x2, y en cada posición de la imagen se aplica una pequeña cuadrícula de parámetros denominada kernel. Un kernel es un extractor de características optimizable,

lo que hace que las CNN sean muy eficientes para el procesamiento de imágenes. En la Figura 5 se muestra la arquitectura de una red neuronal convolucional.

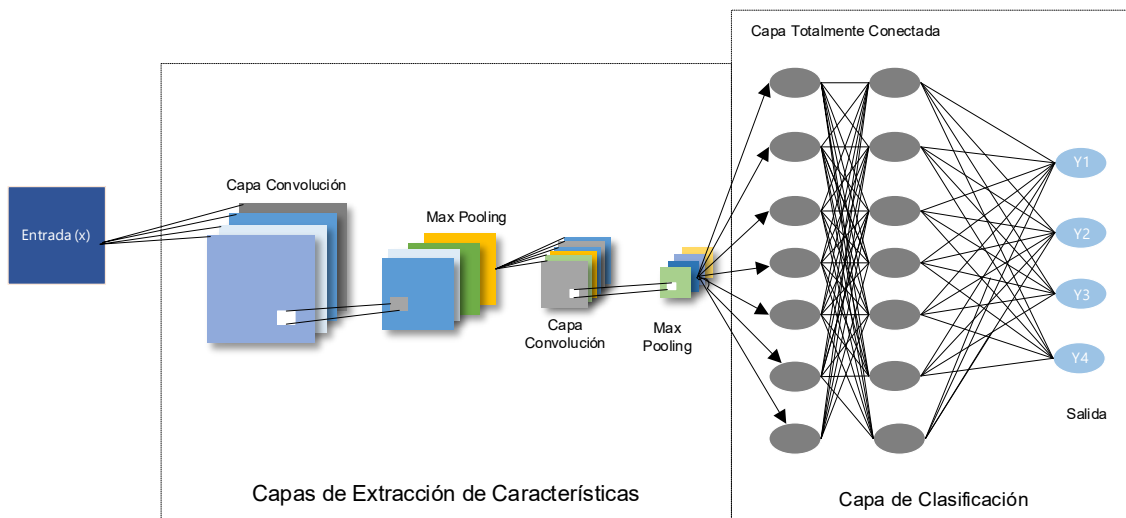


Figura 5. Arquitectura de una red Neuronal Convolucional (CNN). Adaptado de [16]

Es importante rescatar dos aspectos importantes de la CNN. Primero, la CNN no requiere la extracción manual de características. En segundo lugar, la CNN necesita muchos más datos debido a los millones de parámetros que debe estimar y, por lo tanto, es más costosa desde el punto de vista informático, por lo que requiere unidades de procesamiento gráfico, del inglés graphics processing unit, GPU, para el entrenamiento del modelo [16].

1.4.5. Procesamiento de Imágenes

La mayoría de imágenes son tomadas por cámaras de teléfonos móviles o cámaras digitales de baja resolución, generando imágenes con ruido (ruido por iluminación, por sombras entre los objetos, pequeños puntos no deseados dentro de la imagen y otros efectos, como movimiento, etc).

1.4.5.1. Ruido

El ruido es la información no deseada presente en la imagen digital, que provoca una oscilación en el valor de sus píxeles alrededor de su valor real [17]. Generalmente, el ingreso del ruido se produce durante la adquisición y/o transmisión de imágenes por sensores, circuitos de un escáner o una cámara digital [18]. Se pueden encontrar diferentes tipos de ruido, como: ruido gaussiano, ruido de sal y pimienta, ruido de disparo o Poisson y ruido de moteado. Probablemente, el ruido más frecuente es el ruido gaussiano, éste se utiliza ampliamente para modelar el ruido térmico. Además, es independiente en cada píxel e independiente de la intensidad de la señal. Por ejemplo, en

las cámaras de color en las que se utiliza más amplificación en el canal de color azul que en el canal verde o rojo, puede haber más ruido en el canal azul [19].

Por otro lado, una imagen que contenga el ruido de sal y pimienta tendrá píxeles oscuros en regiones brillantes y píxeles brillantes en regiones oscuras, lo cual crea un efecto similar a espolvorear puntos blancos y negros -sal y pimienta- en la imagen. Este tipo de ruido puede ser causado por píxeles muertos, errores del convertidor analógico-digital, errores de bits en la transmisión, etc [18]. Por tanto, es necesario que las imágenes obtenidas “en bruto” sean procesadas para remover dicho ruido.

Existen métodos para eliminar el ruido, resaltar o suprimir información contenida en una imagen a diferentes escalas espaciales, para destacar algunos elementos de la imagen, o también para ocultar valores anómalos. Los métodos más utilizados son los filtros [20].

1.4.5.2. Filtro

Un filtro es un conjunto de operaciones en donde el nuevo valor de un píxel depende de su anterior valor y el de los píxeles vecinos. Este conjunto de operaciones no afecta el tamaño de la imagen resultante, de tal forma que la relación entre la imagen resultante y la imagen original sigue siendo igual que en las operaciones de píxel 1 a 1. El filtro se basa en una función de transferencia que al aplicarse a una señal de entrada, se obtiene una señal de salida según lo que se necesita. Los principales usos de los filtros son para suavizar la imagen que consiste en uniformizar la intensidad de los píxeles vecinos, realzar la imagen que consiste en aumentar los niveles de intensidad, eliminar ruido y detectar bordes [17].

Los filtros aplicados a una imagen pueden ser:

- Filtros de paso bajo (FPB): El término pasa bajo se refiere a que dejar pasar componentes de frecuencia baja y, por el contrario, no deja pasar los componentes de frecuencia alta. En las imágenes resultantes se amortigua los cambios repentinos de niveles de intensidad entre píxeles vecinos. Las principales aplicaciones de este tipo de filtro son la eliminación de ruido, suavizado de la imagen y reducción de la nitidez [17].
- Filtro de paso alto (FPA), El término pasa alto se refiere a que deja pasar los componentes de frecuencia alta y, por el contrario, no deja pasar los componentes de frecuencia baja. La principal aplicación de este tipo de filtro es la detección de zonas de contraste elevado como los bordes [20].
- Filtros Paso Banda: El término pasa banda se refiere a que deja pasar los componentes de cierto rango de frecuencia. Las principales aplicaciones de este tipo de filtro es aislar los efectos de ciertas bandas de frecuencias seleccionadas

sobre una imagen y eliminar ruido, pero son poco usados porque elimina mucho contenido de las imágenes [17].

1.4.5.3. Modelos de Color

Un modelo de color, también llamado espacio de color o sistema de color, es una especificación de un sistema de coordenadas donde cada color está representado por un único punto. Su principal función es especificar colores de una forma estandarizada. Los modelos más utilizados son el modelo RGB (rojo, verde, azul) que es un modelo para monitores de color y una amplia clase de cámaras de vídeo de color, el modelo CMY (cian, magenta, amarillo) y el modelo CMYK (cian, magenta, amarillo, negro) que son los modelos de impresión en color, y el modelo HSI (tono, saturación, intensidad) que es el más cercano a como los seres humanos interpretan el color.

En el procesamiento de imágenes, RGB es el modelo de color más usado en el procesamiento digital de imágenes y aunque el modelo RGB es muy sencillo, poco considera la forma de captura de los colores, además de ser altamente sensitivo al cambio de iluminación [17].

1.4.6. Trabajos Relacionados

Desde hace varios años, se han publicado numerosos trabajos en los cuales se emplea técnicas de aprendizaje de máquina y aprendizaje profundo para clasificar imágenes que se generan en distintas áreas sociales [21]. Por ejemplo, en [22], se compara dos métodos de clasificación: máquina de soporte vectorial y aprendizaje profundo basado en redes neuronales convolucionales para clasificar las imágenes aéreas en 6 categorías, encontradas comúnmente en zonas urbanas. Para efectos de dicho trabajo, la máquina de soporte vectorial implementa descriptores para realizar una tarea de clasificación multiclase; mientras que, la técnica de aprendizaje profundo utiliza cierto porcentaje de las imágenes directamente para aprender y reconocer patrones relevantes en la imagen que se presenta. Así el algoritmo de aprendizaje profundo fue capaz de clasificar imágenes de avenidas, edificios, industrias, zonas naturales, zonas residenciales y cuerpos de agua con una exactitud de 87%. Por otro lado, en [23], se clasificaron las áreas construidas con imágenes HRSS basadas en el aprendizaje profundo, para ello se aplicaron técnicas como redes neuronales convolucionales (CNN), CapsNet, SMDTR-CNN y SMDTR-CapsNet. Estas técnicas se seleccionaron debido a su eficacia y potente rendimiento de identificación. Los resultados de los métodos propuestos se han confirmado en términos de precisión, coeficiente kappa y matriz de confusión, los cuales revelaron que SMDT CNN obtuvo la mejor precisión general (95.0%) y el mejor resultado de coeficiente kappa (0.944).

Por su parte, en [24] se utilizó la inspección humana para clasificar visualmente la calidad de los productos y las materias primas utilizadas en el proceso de producción. Este proceso se automatizó mediante el uso del aprendizaje profundo aplicado en imágenes de tablas de madera. Para ello, los investigadores compararon redes neuronales convolucionales con técnicas de extracción de características basadas en texturas y técnicas tradicionales: algoritmos de inducción de árboles de decisión, redes neuronales, vecinos más cercanos y máquinas de vectores de soporte. Los resultados de dicho trabajo muestran que las técnicas de aprendizaje profundo aplicadas a tareas de procesamiento de imágenes han logrado un rendimiento predictivo superior a las técnicas de clasificación tradicionales. Finalmente, en [25], se aplicó aprendizaje de máquina para inspeccionar piezas de una línea de productos que anteriormente eran inspeccionadas manualmente. En este trabajo, los autores proponen un método para la inspección visual automática de suciedad, arañazos, rebabas y desgastes en las piezas de la superficie. Se aplicó análisis de imágenes con CNN para confirmar la existencia del defecto en la región objetivo de una imagen. Además, los autores construyeron y probaron varios tipos de redes profundas de diferentes profundidades y capas para seleccionar la estructura adecuada para la inspección de defectos en la superficie, concluyendo que, una sola red basada en CNN es suficiente para probar varios tipos de defectos en superficies texturadas y no texturadas. Los trabajos mencionados anteriormente son una muestra de las aplicaciones técnicas de aprendizaje de máquina como ayuda a resolver problemas de clasificación de imágenes. No se encontraron premisas de trabajos relacionados con la clasificación de imágenes de movimientos sísmicos, razón por la cual la contribución del presente proyecto se puede considerar como novedosa.

2. METODOLOGÍA

En este capítulo se describe la metodología seleccionada, “Design Research”, como marco general de investigación [26]. Mediante este enfoque se aplica una aproximación de investigación experimental, que permite examinar o explorar un problema de investigación poco estudiado o que no ha sido analizado antes [27]. Dentro del proceso de implementación del artefacto de investigación, se aplicará el método *Cross Industry Standard Process for Data Mining* (CRISP-DM) para desarrollar el modelo de clasificación de imágenes de fachadas de estructuras de construcción afectadas por un terremoto. CRISP-DM proporciona una descripción normalizada del ciclo de vida de un proyecto estándar de análisis de datos, de forma análoga a como se hace en la ingeniería del software con los modelos de ciclo de vida de desarrollo de software [28].

La metodología CRISP-DM define una secuencia no rígida de seis fases que permiten la construcción e implementación de un modelo de minería de datos en un entorno real, ayudando a apoyar las decisiones empresariales [29].

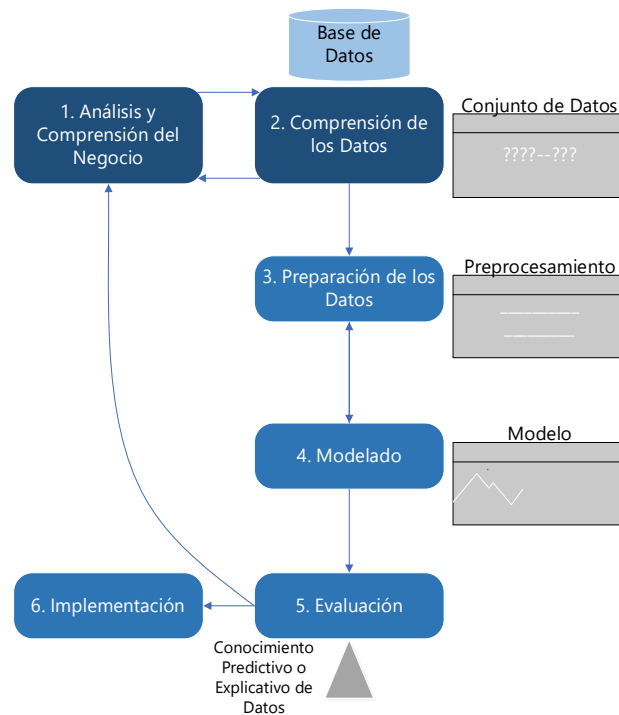


Figura 6. Metodología CRISP-DM. Adaptado de [26].

A continuación, se realiza una descripción breve de la aplicación de las fases de la metodología en este proyecto.

- Análisis y comprensión del negocio: En la primera fase, se busca identificar el objetivo que hay que alcanzar: En este caso, el objetivo es diseñar un modelo de

clasificación supervisada de fotografías de fachadas para evaluar el daño estructural ocasionado por sismos de acuerdo a la EMS para apoyo de toma de decisiones en el IG-EPN, aplicando técnicas de Aprendizaje de Máquina y Aprendizaje Profundo.

- Comprensión de los datos: La segunda fase se refiere recolección, descripción, exploración y verificación de los datos.
- Preparación de los datos: En esta tercera etapa se procesan los datos. Los datos tienen conceptos (lo que hay que aprender), instancias (registros independientes relacionados con una ocurrencia) y atributos (que caracterizan un aspecto específico de una instancia). Posterior a ello, se realiza la fase de modelamiento.
- Modelado: En esta fase se construye el modelo que representa el conocimiento aprendido, es decir el modelo aprende a clasificar las imágenes de acuerdo con el grado de daño [29].
- Evaluación: En esta fase se analiza su validez en la, en términos de su rendimiento y utilidad. Por ejemplo, en las tareas de clasificación (para predecir un objetivo discreto). Las métricas habituales son precisión, recall o f1-score [30]. Ahora bien, si el modelo no es lo suficientemente bueno para utilizarlo como apoyo a la empresa, entonces se define una nueva iteración.
- Implementación: Si el modelo cumple su objetivo se presenta para que se pueda usar en futuras clasificaciones, esto sería la fase de despliegue [29].

En las siguientes subsecciones, desde 2.1 hasta la 2.4, se desarrollan el ciclo propuesto por la Metodología CRISP-DM.

2.1 Análisis y comprensión del negocio

En el Ecuador, desde 1906, cinco terremotos con una magnitud superior a 7,7 se han producido en la parte poco profunda de la zona de subducción, y al menos 13 terremotos destructivos de magnitud menor a 7,7, pero más superficiales y destructivos, que han alcanzado intensidades iguales o superiores a VIII, se han producido en los últimos cinco siglos [31]. Para tener una idea de la frecuencia de sismos grandes (mayores a 6), la Figura 7, muestra que cada quinquenio se puede tener al menos un sismo de magnitud de momento igual o mayor a 7 y cinco o más sismos de magnitud igual o mayor a 6. Por ejemplo, entre 2015 y 2020 existieron 22 eventos mayores o iguales a 6, incluidas varias réplicas del sismo de 7,78 de abril de 2016 [32].

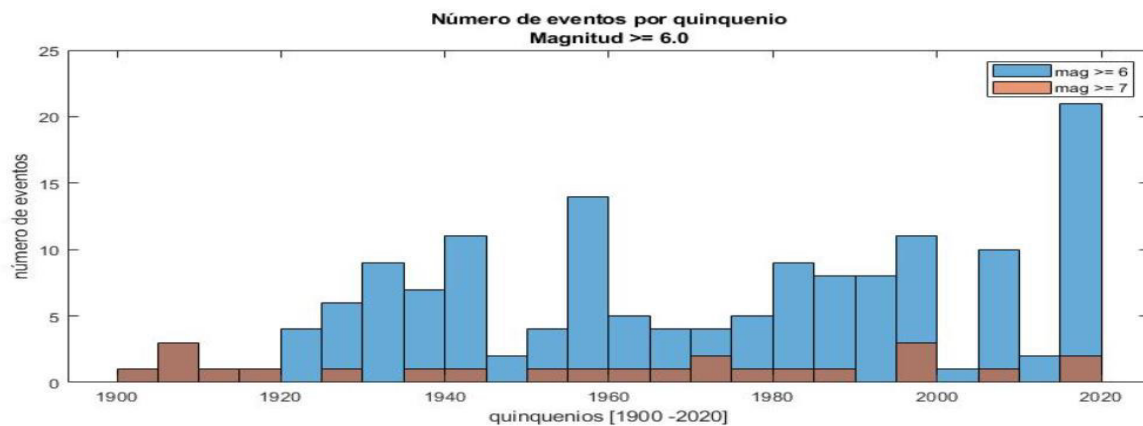


Figura 7. Número de Eventos Sísmicos por Quinquenio y Magnitud ≥ 6.0 [32].

Los daños materiales provocados por estos eventos son imposibles de cuantificar debido al poder destructivo de este tipo de fenómenos naturales [33]. Por tanto, el objetivo de este proyecto es ayudar en la toma de decisiones a los técnicos responsables de la evaluación de la intensidad de un sismo, mediante un modelo de clasificación de imágenes de fachadas de edificios de acuerdo con la EMS, la cual reconoce 5 grados para mampostería y 5 grados para hormigón armado. Se usarán estos dos tipos de estructura debido a que son los más prevalecientes en las urbes. La descripción de cada grado se mostró en la Tabla 1 y 2.

2.2 Comprensión de los datos

Esta fase comprende: la recolección inicial de los datos con el objetivo de establecer un primer contacto con el problema, la familiarización con ellos y un análisis de calidad. Como de costumbre en este tipo de análisis, ésta, junto con las dos siguientes fases, fueron las que demandaron el mayor esfuerzo y tiempo.

2.2.1 Recolección del conjunto de datos

Inicialmente se tenía previsto trabajar juntamente con el IG-EPN para obtener un conjunto de datos etiquetado. Sin embargo, no se pudo obtener un conjunto de imágenes que sirva para el proyecto. Por tanto, se procedió a construir el conjunto de datos desde cero. Para ello se optó por recolectar imágenes de edificios dañados después de un terremoto de varias fuentes como: informes de campos no oficiales, informes oficiales, imágenes de artículos, blogs y periódicos digitales. Un ejemplo de un edificio afectado por un terremoto se muestra en la Figura 8. El listado de las fuentes de datos utilizados se detalla en la Tabla 4.



Figura 8. Terremoto del 10 de diciembre 1970 en Loja [34].

Tabla 4. Fuente de Datos de Imágenes de Edificios con Daños Estructurales

1	Informe del mid-america earthquake center sobre el sismo de pisco –chinchá, 15/08/07
2	Informe de la base de datos “Daños Sísmicos en Construcciones”: Terremoto de Alhucemas.
3	El mega sismo de Chile de 2010 y lecciones para Ecuador.
4	Evaluación del riesgo sísmico de edificios existentes en la Isla Isabela, Galápagos
5	Estudio de la zona afectada por el terremoto de Pisco, Misión Ceresis/ Unesco, 15/08/2007
6	Intensidades sísmicas en el área de daños del terremoto del 27 de febrero de 2010, Santiago-Chile
7	The Mw 8.8 Chile Earthquake of February 27, 2010
8	Effects of Earthquakes on Buildings in the Ibero-Maghrebian Region
9	Damage Assessment and Seismic Intensity Analysis of the 2010. Mw 8.8 Maule Earthquake
10	Earthquake –Resistant Design Concepts
11	Seminario de promoción de la normativa para el diseño y construcción de edificaciones seguras
12	Intensidad Macrosísmica Aránzazu Izquierdo Álvarez
13	Necesidad de reforzar las estructuras afectadas por un terremoto
14	Inspección y evaluación rápida de estructuras Post-evento Sísmico
15	https://www.bbc.com/news/world-europe-37176601?fbclid=iwar2fnaewmcpn6xzw3tikldnbnxagihgkanuu66b4d_idb3fv9rgjproxbc
16	https://www.outlookindia.com/photos/topic/earthquake/100839?photo-281750
17	https://www.ngdc.noaa.gov/hazardimages/?fbclid=iwar0rbx3bzc2ocifyfzkvyracs7w9wurfujrc_mrtajwnlcqz_sk5ny2jnva#/earthquake
19	https://www.emsc-csem.org/earthquake/gallery/?fbclid=iwar1gewlx0rgl8pe46czg3j5fk-nf_xfbkdnfcqtmboafv0f_5twhwfhklmi
20	https://www.gettyimages.com.mx/fotos/house-earthquake?page=2&phrase=house%20earthquake&sort=mostpopular
21	https://sp.depositphotos.com/stock-photos/earthquake-house.html
22	https://www.usgs.gov/natural-hazards/earthquake-hazards/science/earthquake-photo-collections?qt-science_center_objects=0#qt-science_center_objects
23	https://www.bbc.com/mundo/noticias-america-latina-58290240
24	https://www.google.com/search?q=terremoto+haiti+2021&sxsrf=aoaemvlpj0cvii0bxjce3alc9wkn6idgg:1630884937610&source=lnms&tbn=isch&sa=x&ved=2ahukewigtidr_-jyahu1sjabhryicdyq_auoaxoecaeqaw&biw=1600&bih=777
25	https://www.elnuevotiempo.com/terremoto-de-7-2-sacude-haiti-y-deja-centenares-de-muertos/
26	https://www.dw.com/es/terremoto-deja-dolor-y-casi-1300-muertos-en-haiti/a-58874177
27	https://saludpublica.mx/index.php/spm/article/view/9408/11478
28	https://www.washingtonpost.com/es/post-opinion/2019/09/19/dos-anios-del-sismo-en-cdmx-no-hemos-aprendido-nada/
29	http://www.cconstruccion.net/eventos_files/doc/WalterMera.pdf
30	https://www.eldiario.ec/especial/365dias-Terremoto-Ecuador/hotelaria/1
31	https://es.123rf.com/imagenes-de-archivo/earthquake_broken.html?oriSearch=earthquake&page=5&sti=o8qn07ospfpbs90oph%7C
32	https://www.istockphoto.com/es/search/2/image?family=creative&mediatype=photography&phrase=earthquake

2.1.1 Descripción del conjunto de datos

El conjunto de datos creado consta de 392 imágenes de mampostería (MA) y 213 imágenes de hormigón armado (HA), sumando un total de 605 imágenes que hacen referencia a terremotos ocurridos en Croacia (2020), Turquía (2020), Albania (2019), Filipinas (2019), México (2017), Ecuador (1970, 2016), Italia (2016), Japón (2011), Chile (2010). Además, se encontraron imágenes de terremotos históricos como el de Chile (1960, 1985), Colombia (1999). Con dichas imágenes se organizó el conjunto de datos en 2 directorios (mampostería y hormigón), cada directorio contiene 5 subdirectorios, cada subdirectorio pertenece a un grado de daño, los cuales se detallaron en las Tablas 1 y 2.

La Tabla 5 muestra la cantidad de imágenes extraídas por tipo de estructura y por grado para el estudio.

Tabla 5. Distribución del Conjunto de Datos según el Grado y Tipo de Estructura

Conjunto de datos	Mampostería	Hormigón Armado	Total
Grado 1	58	50	108
Grado 2	124	43	167
Grado 3	47	35	82
Grado 4	92	41	133
Grado 5	71	44	115
Total	392	213	605

2.1.2 Exploración del conjunto de datos

El conjunto de datos construido tiene imágenes en bruto con una gran variedad de tamaños y formatos (bpm, tiff, png, jpg), en escala de grises y a color. Por lo general, hay que procesar los datos en bruto para poder introducirlos en una red neuronal [13].

2.3 Preparación de los datos

En esta fase y una vez efectuada la recolección de los datos se procede a la preparación de los mismos. Las tareas generales que se incluye en esta fase son: selección de datos, limpieza de datos, construir los datos, integrar los datos y formatear los datos [28].

2.3.1 Selección de datos

En esta fase se selecciona un subconjunto de los datos adquiridos. Para ello, se inicia con la clasificación manual de las imágenes mediante la observación de sus características físicas [8] y en base a las ilustraciones y descripciones de la EMS (Tablas 1 y 2 del Capítulo 1).

A continuación, la Tabla 6 presenta una muestra de imágenes dado su tipo de estructura y categorizadas por el grado de daño. Por ejemplo, las imágenes **A** y **B** han sido clasificadas en el grado 1 debido a que no tienen ningún daño estructural, solamente pequeñas fisuras en sus muros. La imagen **C** ha sido clasificada en el grado 2 de mampostería debido a que tiene grietas en todo el muro frontal de la casa, mientras que la imagen **D** tiene caída de enlucido y revestimientos frágiles en la parte superior del edificio. Esto representa daño estructural ligero.


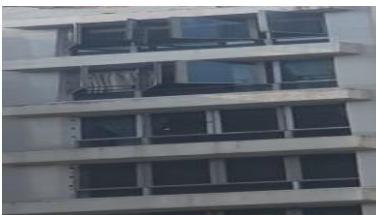








Por otro lado, la imagen **E** se encuentra en la categoría de grado 3 debido a que el tejado está caído; y la imagen **F**, es un edificio con el mismo grado de daño 3 ya que presenta desprendimiento de hormigón en varios pisos del edificio, dejando ver las columnas del edificio, esto significa que las imágenes E y F tienen daños no estructurales graves.

Además, la imagen **G** pertenece al grado 4 de daño de tipo mampostería debido a que tiene dañado parcialmente el techo y el piso, lo cual representa daño no estructural muy grave. La imagen **H** también pertenece al grado 4 de tipo hormigón armado ya que se observa que la mayoría de pisos superiores del edificio colapsaron, por tanto, significa un daño estructural grave.

Finalmente, en la imagen **I** es fácil distinguir que ha colapsado totalmente. Esto representa daños estructurales muy graves, por consiguiente pertenece al grado 5 de daño de tipo mampostería, mientras que, en la imagen **J** se puede evidenciar que el edificio ha colapsado de un ala del mismo, por tanto, se produce un ladeo. Esto da lugar a un daño estructural muy grave, por tanto se asigna un grado de daño 5 de tipo hormigón armado.

Es importante, recalcar que en este trabajo se han excluido imágenes de los interiores de los edificios, así como también, imágenes de columnas destruidas, es decir, que el lente de la cámara haya captado sólo la columna interna o externa de un edificio, solamente se incluye imágenes de las fachadas de los edificios. Además, para mejorar las condiciones de partida de la imagen y reducir el espacio de búsqueda con alta probabilidad de contener individualmente características que aporten información al modelo, se obtiene un conjunto reducido de subimágenes (candidatas), el cual se detalla en la siguiente sección.

Tabla 6. Clasificación Manual de Imágenes

Grado	Mampostería	Hormigón armado
Grado1	<p>A</p> 	<p>B</p> 
Grado2	<p>C</p> 	<p>D</p> 
Grado3	<p>E</p> 	<p>F</p> 
Grado4	<p>G</p> 	<p>H</p> 
Grado5	<p>I</p> 	<p>J</p> 

2.3.2 Construcción del conjunto de datos

Para la construcción del conjunto de datos que es usado para entrenar el modelo, se procede a estandarizar el tamaño, y el formato de las imágenes.

2.3.2.1 Formato de imágenes

De acuerdo al tema de investigación, las imágenes pueden ser almacenadas en diversos formatos y, en diferentes resoluciones. Para almacenarlas con una buena resolución debe existir un equilibrio entre la calidad y el tamaño. Sin embargo, el tamaño de cada imagen es afectado por el formato del mismo. Actualmente, existen distintos métodos de compresión que se usan en los formatos de archivo, entre ellos, JPG, PNG, BMP, TIFF. Dentro de esos formatos de archivo, BMP y TIFF, ofrecen excelente calidad y resolución. Sin embargo, necesitan mayor espacio en disco de almacenamiento y requieren más tiempo para editar e imprimir, a diferencia de los formatos de archivo JPG, o PNG [35].

En la Figura 9, se muestra un ejemplo con diferente formato de archivo y su espacio ocupado en disco, siendo el formato PNG el que menor espacio de almacenamiento ocupa en comparación con los demás formatos, seguido de JPG. Además, en la Tabla 7 y 8 se muestra la distribución del conjunto de datos de tipo mampostería y hormigón según el formato de archivo.

Con base a lo mencionado anteriormente, se decide trabajar en este proyecto con el formato de archivo .jpg y también porque es el formato que más se usa en el conjunto de datos.





 1_ejemplo	07/11/2021 22:02	Archivo BMP	79 KB
 1_ejemplo	07/11/2021 22:03	Archivo TIFF	44 KB
 1_ejemplo	29/10/2021 12:15	Archivo JPG	25 KB
 1_ejemplo	07/11/2021 22:00	Archivo PNG	13 KB

Figura 9. Tamaño de una misma imagen con diferentes tipos de formatos.

Tabla 7. Distribución del Conjunto de Datos de Tipo Mampostería según el Formato.

Conjunto de datos	Mampostería				Total
	TIFF	JPG	PNG	JPEG	
Grado 1	8	23	12	15	58
Grado 2	12	77	25	10	124
Grado 3	10	18	12	7	47
Grado 4	30	24	24	14	92
Grado 5	12	24	16	19	71
Total	72	166	89	65	392

Tabla 8. Distribución del Conjunto de Datos de Tipo Hormigón Armado según el Formato

Conjunto de datos	Hormigón				Total
	TIFF	JPG	PNG	JPEG	
Grado 1	10	14	16	10	50
Grado 2	14	9	15	5	43
Grado 3	5	11	10	9	35
Grado 4	14	16	5	6	41
Grado 5	13	15	10	6	44
Total	56	65	56	36	213

2.3.2.2 Tamaño de imágenes

La mayoría de modelos de clasificación de imágenes de Keras, por ejemplo, la familia de MobileNets, han sido entrenadas con imágenes de tamaño 224x224x3, el último dígito de la estructura hace referencia al número de canales, es decir, codificación RGB (red, green, blue). Además, dichos modelos admiten cualquier tamaño de entrada superior a 32 x 32. Es importante señalar que a pesar de que los tamaños de imagen más grandes ofrecen un mejor rendimiento [36], también ocupan mayor espacio en disco de almacenamiento. Por tanto, se debe encontrar un balance entre estos dos aspectos.

En este proyecto se establece el tamaño de imagen en 100x100 y se procesa el conjunto de datos con una codificación RGB.

2.3.2.3 División del conjunto de datos

En la Tabla 4 se indicó el número total de imágenes obtenidas de acuerdo al tipo de estructura y grado de daño. Ahora bien, de acuerdo al tipo de estructura se forman dos conjuntos de datos, mampostería (MA) y hormigón armado (HA). A su vez, los conjuntos de datos, MA y HA, se subdividen en datos de entrenamiento (80%), validación (10%) y prueba (10%), respectivamente.

2.4 Preprocesamiento de los datos

El procesamiento de imágenes es aplicado cuando resulta necesario modificar una imagen para mejorar su apariencia y así destacar algún aspecto de la información que se encuentra contenida. En esta sección se menciona las técnicas empleadas para el mejoramiento de la imagen.

2.4.1 Conversión de una imagen a color RGB

En el proceso de captación de imágenes, se obtuvo imágenes de color y en escala de grises. Entre los espacios de color más utilizados para el procesamiento de imágenes están

el rojo, verde y azul (RGB). Sin embargo, la elección de un espacio de color depende del uso que se desea para la imagen. Por ejemplo, en caso de que fuera necesario hacer un análisis en relación con la superficie, se utilizará la combinación CMYK Cyan, Magenta Yellow Black que tiene mucha utilidad para la comparación de patrones de impresión. Mientras que, para el campo de procesamiento y análisis de píxeles en una imagen se utiliza el espacio de color RGB. Cada uno de estos tres colores está representado por un número entero comprendido en el rango de 0 a 255. Dado que el valor de un píxel solo necesita estar en el rango [0, 255] se usa normalmente un entero sin signo de 8 bits para representar cada intensidad de color [35].

En este caso, las imágenes se procesan en 3 canales (RGB), debido a que los modelos preentrenados de que se usaran en este proyecto usan este tipo de color de imagen. Dichos modelos se describen en la sección 2.5.1.

2.4.2 Suavizado de la imagen

Se hicieron pruebas con diferentes tipos de filtros, filtro promedio como se muestra en la Figura 10, filtro gaussiano que se indica en la Figura 11, filtro de mediana se visualiza en la Figura 12, pero el filtro con el que se obtuvieron mejores resultados para este proyecto fue el filtro promedio ya que, además de eliminar el ruido, realiza un suavizado de la imagen.



Figura 11. Filtro Gaussiano



Figura 10. Filtro Promedio



Figura 12. Filtro De Mediana

2.4.3 Aumento de datos

El procesamiento de imágenes es uno de los ámbitos en los que el aumento de datos es muy adecuado. Esto se debe a que muchas transformaciones, como la traslación, la rotación, la extracción de parches y la reflexión, no cambian fundamentalmente las propiedades del objeto en una imagen [37]. Por tanto, el aumento de datos ayuda a mejorar la precisión del modelo, la generalización y a controlar el sobreajuste [38].

Para aplicar esta técnica se utilizó el paquete de Python, Augmentor. Entre sus principales características se tiene la rotación (en inglés *rotation*), cizallamiento (en inglés *shearing*), recorte (en inglés *cropping*), reflejo (en inglés *mirror*) [38], entre otras. Aunque estas funcionan mejor en algunos ámbitos que en otros [37].

En este trabajo, se aplicó reflejo o *shearing*, y las demás se descartaron, debido a que, por ejemplo, si se aplicaba rotación a un edificio que solo sufrió daños leves, no estructurales, y que se mantenía en pie, se podría confundir con alguna otra que, producto de un sismo, sí se cayó o giró totalmente. En la Figura 13, se muestra un ejemplo de esta técnica aplicada a un edificio que sufrió daños de grado 4.

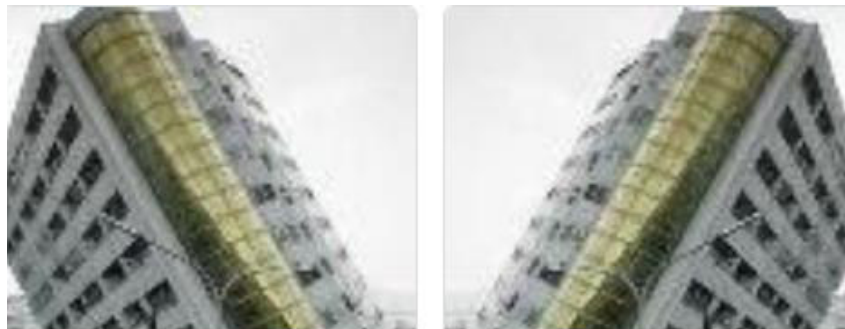


Figura 13. Técnica de Aumento de Datos aplicado a un Edificio.

Producto de aplicar esta técnica se generaron dos nuevos conjuntos de datos, Mampostería 1 (MA_1) y Hormigón Armado (HA_1).

2.5 Modelado

En esta sección se describen las tareas que permitieron crear el modelo.

2.5.1 Selección de herramientas

Las herramientas empleadas para el desarrollo del modelo fueron:

- Debido a la falta de un ordenador con recursos computacionales de gran rendimiento, este trabajo se ha implementado en el entorno Google Colaboratory, también llamado Colab. Este producto de Google Research es ideal para aplicarlo

en proyectos de aprendizaje automático, análisis de datos y educación. Además, brinda la posibilidad de utilizar tanto CPUs como GPUs [39].

- Python es uno de los lenguajes de programación más populares para la ciencia de datos y, gracias a su comunidad de desarrolladores y de código abierto, se ha desarrollado un gran número de bibliotecas útiles para la computación científica y el aprendizaje automático [40].
- Tensorflow es una librería de código abierto desarrollado por Google y fácil de integrar con proyectos realizados en Python [40].
- Keras es un framework de alto nivel, es una API escrito en lenguaje Python y creado sobre Tensorflow para la construcción de redes neuronales [41].
- Scikit Learn es la biblioteca más útil y robusta para el aprendizaje automático en Python. Ésta proporciona herramientas eficientes para el aprendizaje automático y el modelado estadístico, incluyendo clasificación, regresión, clustering y reducción de dimensionalidad a través de una interfaz consistente en Python [42]. Además, es ampliamente utilizada en varios contextos. La biblioteca está construida sobre NumPy, SciPy y Matplotlib. Es de código abierto, utilizable comercialmente bajo licencia BSD, *Berkeley Software Distribution* [43].

2.5.2 Arquitectura de la red neuronal convolucional

2.5.2.1 Aprendizaje por transferencia

En este trabajo no se ha desarrollado una CNN propia debido a que el conjunto de datos es escaso para entrenar el modelo desde cero. Por tanto, se optó por utilizar el Aprendizaje por Transferencia o en inglés conocido como Transfer Learning (TL), el cual consiste en tomar las características aprendidas en un problema y aprovecharlas en un nuevo problema similar [44]. Las arquitecturas de este tipo han sido entrenadas con un enorme conjunto de imágenes, regularmente con numerosas clases de objetos cotidianos. Las arquitecturas que han demostrado solvencia en la resolución de problemas complejos y que contribuyen al avance del aprendizaje profundo son: VGG16, InceptionResnetV2 (híbrida de las arquitecturas Inception y Resnet), MobileNetV2, VGG16, DenseNet121, DenseNet201, Xception, ResNet50, VGG19, InceptionV3, y ResNet152V2 (Anexo 1). [45], entre otras. A continuación, se describen brevemente los modelos mencionados anteriormente.

La red VGG, *Visual Geometry Group*, es un modelo de CNN ideado por Simonyan y Zisserman (2015) para el ImageNet Large Scale Visual Recognition Challenge 2014 (ILSVRC-2014). El modelo alcanzó una tasa de error del 7,5 % en el conjunto de validación,

resultado que les aseguró el segundo puesto en la competición [46]. Su mejora se basa en el uso de núcleos pequeños ($f=3$) que permiten configuraciones más profundas sin comprometer el número de parámetros y, por lo tanto, la eficacia del entrenamiento. Una innovación importante de VGG es que redujo el tamaño de los filtros, pero aumentó la profundidad. Es importante entender que la reducción del tamaño de los filtros requiere un aumento de la profundidad. Esto se debe a que un filtro pequeño sólo puede capturar una pequeña región de la imagen a menos que la red sea profunda. El aumento de la profundidad también permite utilizar un mayor número de activaciones no lineales, lo que aumenta el poder de discriminación del modelo. Por lo tanto, el VGG siempre utiliza filtros con huella espacial 3×3 y pooling de tamaño 2×2 . Esta elección de diseño da como resultado un cierto nivel de equilibrio en el esfuerzo computacional a través de las capas, y fue heredada por algunas de las arquitecturas posteriores como ResNet [47]. DenseNet121 propaga la salida de cada capa a todas las capas posteriores. Para ello, es necesario concatenar los mapas de características de una capa con todos los mapas de características que la preceden. Esto permite el paso de información sin ninguna pérdida de información. Tal y como afirman los autores del modelo, la información se va lavando a través del procesamiento de cada capa convolucional y de pooling, de ahí la necesidad de utilizar mapeos de identidad para pasar la información original. Para mejorar la compacidad del modelo, Huang introdujo la capa de transición, en la que se implementa una combinación de convoluciones 1×1 , normalización por lotes y un pooling promedio 2×2 . Además, los mapas de características concatenados de las capas anteriores no son sintonizables. Sólo se utiliza la información global de cada etapa. Por ello, cada capa aprende su propia representación del problema. A pesar de la arquitectura densa de DenseNet, la versión de 121 capas sólo necesita 8 millones de parámetros para entrenarse, lo que supone una mejora en comparación con otros modelos [46].

La arquitectura Inception, introducida por Szegedy en 2014 como GoogLeNet (Inception V1), perfeccionada posteriormente como Inception V2, Inception V3, y más recientemente Inception-ResNet [48], Desde su introducción, Inception ha sido una de las familias de modelos con mejor rendimiento en el conjunto de datos ImageNet [49], así como en conjuntos de datos internos utilizados en Google, en particular JFT [50]. El bloque fundamental de los modelos de estilo Inception es el módulo Inception, del que existen varias versiones diferentes. Esta red ha sido preentrenada en ImageNet. Además, el tamaño de entrada definido para este modelo es de 299×299 en tres canales (RGB) [51]. InceptionResNetV1, es una combinación híbrida entre Inception y ResNet. El enfoque Inception postula la aplicación de convoluciones de distintos tamaños en la misma capa, lo cual aporta flexibilidad en la detección de patrones que pueden aparecer con muy diversas

dimensiones. Estos postulados impresionan favorecer el reconocimiento de aceitunas, independientemente de su tamaño. Por su lado, el paradigma ResNet aporta a la red la potencia del tratamiento de los residuos de aprendizaje. Para una capa dada, el residuo es básicamente la diferencia entre lo aprendido al inicio y al final de esta. La arquitectura ResNet explota este residuo en su misma topología para favorecer y optimizar la convergencia a la solución óptima. Es una red de 164 capas y 55,9 M de parámetros. Ello, junto con su arquitectura híbrida anteriormente discutida, le confiere una gran potencia de razonamiento [52]. DenseNet201 requieren menos parámetros que una CNN tradicional equivalente, ya que no hay necesidad de aprender mapas de características redundantes. Además, algunas variantes de ResNets han demostrado que muchas capas apenas contribuyen y pueden eliminarse. De hecho, el número de parámetros de las ResNets es grande porque cada capa tiene sus pesos que aprender. En cambio, las capas de las DenseNets son muy reducidas (por ejemplo, 12 filtros), y sólo añaden un pequeño conjunto de nuevos mapas de características. Otro problema de las redes muy profundas eran los problemas para entrenar, debido al mencionado flujo de información y gradientes. Las redes densas resuelven este problema, ya que cada capa tiene acceso directo a los gradientes de la función de pérdida y a la imagen de entrada original [53]. Xception supera ligeramente a Inception V3 en el conjunto de datos ImageNet (para el que se diseñó Inception V3), y supera significativamente a Inception V3 en un conjunto de datos de clasificación de imágenes más amplio que comprende 350 millones de imágenes y 17.000 clases. Dado que la arquitectura de Xception tiene el mismo número de parámetros que Inception V3, las ganancias de rendimiento no se deben a un aumento de la capacidad, sino a un uso más eficiente de los parámetros del modelo. ResNet50, abreviatura de Residual Networks, es una red neuronal clásica utilizada como columna vertebral para muchas tareas de visión por ordenador. La red residual de Microsoft introdujo lo que se conoce como el bloque residual, que permitía entrenar redes más profundas (hasta 152 capas) superando el problema de la saturación de la precisión debido a la incapacidad de propagar gradientes. El bloque residual utiliza conexiones de acceso directo (o conexiones de salto) en las que el gradiente se transmite directamente a las capas más profundas. De hecho, la conexión de salto añade las salidas de las capas apiladas a un mapeo de identidad de la o las capas anteriores. Al igual que Google net, esta red utiliza los bloques residuales como bloques de construcción apilados. Este diseño ha alcanzado un 3,57% de error máximo en ImageNet, ganando el concurso en 2015. Para entrenar el modelo ResNet 50 se necesita un total de 25 millones de parámetros [54]. ResNetNet152V2 utiliza 152 capas, lo que supone casi un orden de magnitud más que las utilizadas anteriormente por otras arquitecturas. Esta arquitectura fue la ganadora del concurso ILSVRC en 2015, y

logró un error en el top-5 del 3,6%, lo que dio lugar al primer clasificador con un rendimiento de nivel humano. Esta precisión se consigue con un conjunto de redes ResNet; incluso un solo modelo logra una precisión del 4,5% [37]. El modelo VGG-19 tiene aproximadamente 143 millones de parámetros, donde los parámetros se aprenden a partir del conjunto de datos de ImageNet que contiene 1,2 millones de imágenes de objetos generales de 1.000 categorías de objetos diferentes para el entrenamiento. El VGG-19 contiene 19 capas entrenables, incluyendo capas convolucionales y totalmente conectadas, así como max pooling y dropout [37]. La arquitectura de la CNN VGG-19 se ha reportado para lograr altas precisiones para el procesamiento de imágenes de grandes conjuntos de datos como ImageNet [55]. Estas CNN han sido entrenadas con millones de imágenes; principalmente con el conjunto de datos más usado es ImageNet, el cual contiene 14.197.122 imágenes y 1000 clases [56] y aunque, las versiones disponibles no hayan sido entrenadas para el caso de interés requerido, el conocimiento que proporcionan estas redes es de gran valor para enfrentar nuestro problema.

2.5.2.2 Arquitectura usada

MobileNet es uno de los modelos de aprendizaje profundo que Keras Applications pone a disposición junto con sus pesos preentrenados y que sirven para la predicción, la extracción de características y ajuste. Los modelos de MobileNet regularmente se utilizan en aplicaciones de visión móvil e integrada. Este tipo de modelo se basa en convoluciones separables en profundidad, que es una forma de convoluciones factorizadas que factorizan una convolución estándar en una convolución en profundidad y una convolución 1×1 llamada convolución en puntos. En el caso de MobileNets, la convolución en profundidad aplica un único filtro a cada canal de entrada y la convolución puntual aplica una convolución 1×1 para combinar los resultados de la convolución en profundidad. Una convolución estándar filtra y combina las entradas en un nuevo conjunto de salidas en un solo paso; mientras que, la convolución separable en profundidad divide esto en dos capas, una capa separada para filtrar y otra para combinar. Esta factorización tiene el efecto de reducir drásticamente el cálculo y el tamaño del modelo [57]. Es por ello que, en este proyecto, se empleará el modelo MobileNetV2. MobileNetV2 es muy similar a la MobileNet original, excepto que utiliza bloques residuales invertidos con características de cuello de botella. Además, tiene un número de parámetros drásticamente inferior al de la MobileNet original. Así, es posible utilizar el aprendizaje por transferencia para personalizar el modelo MobileNetV2 y resolver nuestra problemática.

Generalmente, el flujo de trabajo de aprendizaje automático por transferencia consiste en examinar y comprender los datos y pre-procesar los datos (estas actividades fueron

descritas en secciones anteriores). Luego de ello, se carga el modelo base preentrenado (y pesos preentrenados), se apila las capas de clasificación en la parte superior. Finalmente, se entrena y evalúa el modelo. Este flujo se aplica para el conjunto de datos MA y HA.

2.5.2.3 Entrenamiento y optimización de la red neuronal convolucional

En esta fase se hace uso de pesos preentrenados sobre el conjunto de datos ImageNet. Para entrenar la red es necesario establecer el número de épocas y el tamaño de batch que se utiliza. El número de épocas indica el número de veces que el conjunto completo de los datos es pasado a través de la red, mientras que el tamaño de batch determina el número de imágenes que se utilizan para calcular los gradientes necesarios para actualizar los pesos de la red.

En este caso se ha decidido utilizar un tamaño de batch de 64 para la red. Ésto permite que se cargue en memoria ese número de imágenes al mismo tiempo. Por otra parte, el número de épocas suele ser un dilema siempre, ya que usar demasiadas épocas puede llevar a un sobreajuste (overfitting), o usar pocas épocas puede dar lugar a un modelo infra ajustado (underfitting). Una solución a tal dilema es la parada temprana, conocida como early stopping, para ello, keras ofrece el método EarlyStopping, que permite detener el entrenamiento una vez que el rendimiento del modelo deja de mejorar [37]. Es por esto que, luego de pequeños experimentos, el número de épocas a emplear se determinó en 100.

Además, se codificaron las etiquetas a través de la codificación categórica, también conocida como codificación one-hot que se representa en el Anexo 1 y se utilizó categorical_crossentropy como función de pérdida. Finalmente, el algoritmo de optimización que ajustó los pesos de la red fue Adam (Adaptative Moment Estimation) [58], el cual combina los beneficios de otros algoritmos y técnicas de optimización [59], como son Momentum [60] y RMSprop [61] y termina finalmente con una capa densa de tamaño N igual a 5 como se observa en el Anexo 2.

2.6 Evaluación del modelo

Se pueden emplear múltiples herramientas para la interpretación de los resultados. Por ejemplo, las matrices de confusión se emplean en problemas de clasificación y consisten en una tabla que indica cuantas clasificaciones se han hecho para cada tipo. La diagonal de la tabla representa las clasificaciones correctas. Si el modelo generado es válido en función de los criterios de éxito establecidos en la fase anterior, se procede a la explotación del modelo [62]. En la sección 2.3.2.3 se dividieron los dataset MA y HA en tres subconjuntos: entrenamiento, validación y prueba. Este último permitirá evaluar el modelo. Para ello, se realiza la predicción en dichos datos y se compara con las etiquetas reales. A partir de ello se obtiene, además de la matriz de confusión, la precisión, recall y F1-Score. Los resultados obtenidos en esta fase se muestran en el Capítulo 3.

3. RESULTADOS Y DISCUSIÓN

En esta sección se muestran los resultados obtenidos del modelo, producto del entrenamiento sobre los dos conjuntos de datos, MA y HA. Estos conjuntos de datos contienen los grados de daños de estructuras de edificios, los cuales están descritos en la sección 2.5.2.2. Luego de ello se procedió a realizar las respectivas comparaciones y resúmenes para su discusión.

En este experimento se amplió el número de ConvNets pre-entrenadas a un total de diez modelos. Estos modelos corresponden a la mitad de los veinte modelos disponibles en Tensorflow Keras [36]. Los diez modelos seleccionados alcanzan una precisión (accuracy) entre el 71% y 95% sobre el conjunto de datos de ImageNet. Dichos modelos son: MobileNetV2, VGG16, DenseNet121, InceptionResNetV2, DenseNet201, Xception, ResNet50, VGG19, InceptionV3, y ResNet152V2 (Anexo 1).

Cabe recalcar que el fin de este estudio es realizar una comparación del comportamiento de cada modelo seleccionado frente a otro, mas no determinar un ganador o perdedor, es decir, el objetivo es adquirir y ampliar el conocimiento de la utilidad que ofrecen las redes neuronales convolucionales y seleccionar el modelo o los modelos que mejor se comportan sobre el conjunto de datos de este estudio.

A continuación, se señalan los resultados alcanzados durante la etapa de entrenamiento, validación y prueba de los modelos, con y sin aumento de datos. Para ello, se utilizan cuatro conjuntos de datos, MA, HA, MA₁, y HA₁. Cabe recordar que MA₁, y HA₁ son producto de aplicar la técnica de aumento de datos, la cual se mencionó en la sección 2.4.3.

Además, se indican los resultados de las métricas de clasificación, precision, recall y F1-Score. Finalmente, se presenta la matriz de confusión de los mejores modelos, es decir, de aquellos modelos cuyo rendimiento supere el 85% en la métrica de precisión.

3.1. Resultados de entrenamiento con aprendizaje por transferencia (sin aumento de datos).

En esta sección se utilizan los conjuntos de datos, MA y HA, para probar el rendimiento de aprendizaje por transferencia sin aumento de datos. Durante el proceso se obtuvieron los resultados que se muestran en las Tablas 9 y 10. En dichas tablas se muestra la exactitud en la etapa de entrenamiento, validación y prueba, y el número de épocas que se ejecutaron en cada modelo.

Tabla 9. Resultado de entrenamiento, Validación, y Prueba MA

Modelo	Mampostería (MA)			Épocas (/100)
	Exactitud			
	Entrenamiento	Validación	Prueba	
MobileNetV2	0,9960	0,8254	0,8734	9
VGG16	1,0000	0,746	0,8101	56
DenseNet121	1,0000	0,7143	0,7975	27
InceptionResNetV2	1,0000	0,6667	0,7595	86
DenseNet201	1,0000	0,7619	0,8228	26
Xception	1,0000	0,6825	0,7848	45
ResNet50	0,6160	0,6032	0,6203	20
VGG19	1,0000	0,7619	0,8608	51
InceptionV3	1,0000	0,6032	0,7722	60
ResNet152V2	1,0000	0,7937	0,7468	22

Tabla 10. Resultados de Entrenamiento, Validación, y Prueba HA

Modelo	Hormigón Armado (HA)			Épocas (/100)
	Exactitud			
	Entrenamiento	Validación	Prueba	
MobileNetV2	1,0000	0,7059	0,7209	17
VGG16	1,0000	0,7647	0,7647	54
DenseNet121	1,0000	0,7059	0,6512	24
InceptionResNetV2	1,0000	0,6471	0,5581	50
DenseNet201	1,0000	0,7647	0,7442	25
Xception	1,0000	0,7059	0,6047	42
ResNet50	0,3603	0,5294	0,4419	13
VGG19	1,0000	0,7647	0,7442	49
InceptionV3	1,0000	0,6471	0,5349	71
ResNet152V2	1,0000	0,7353	0,6047	18

Los resultados obtenidos de las Tablas 9 y 10 no son suficientes para elegir al modelo más óptimo. Por tanto, en las Figuras 7 y 8, se muestra una comparativa de la métrica de exactitud (accuracy) entre los diez modelos seleccionados para los conjuntos de datos, MA y HA.

Específicamente, en la Figura 14 se evidencia que MobileNetV2, VGG19, DenseNet201, VGG16 y DenseNet121 tienen un rendimiento superior al 80% en el conjunto de datos MA, de los cuales solo MobileNetV2 y VGG19 superan el 85%. Por otro lado, en la Figura 15 se indica que MobileNetV2, VGG19, DenseNet201, y VGG16 alcanzan un rendimiento entre el 70% y 75% para el conjunto de datos de HA, pero ninguno alcanza el 85%.

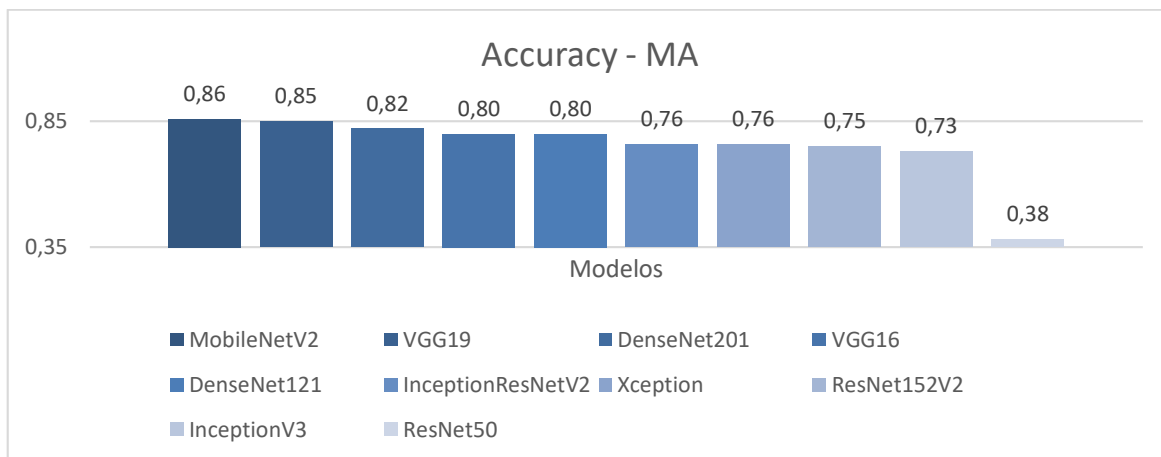


Figura 12. Accuracy MA

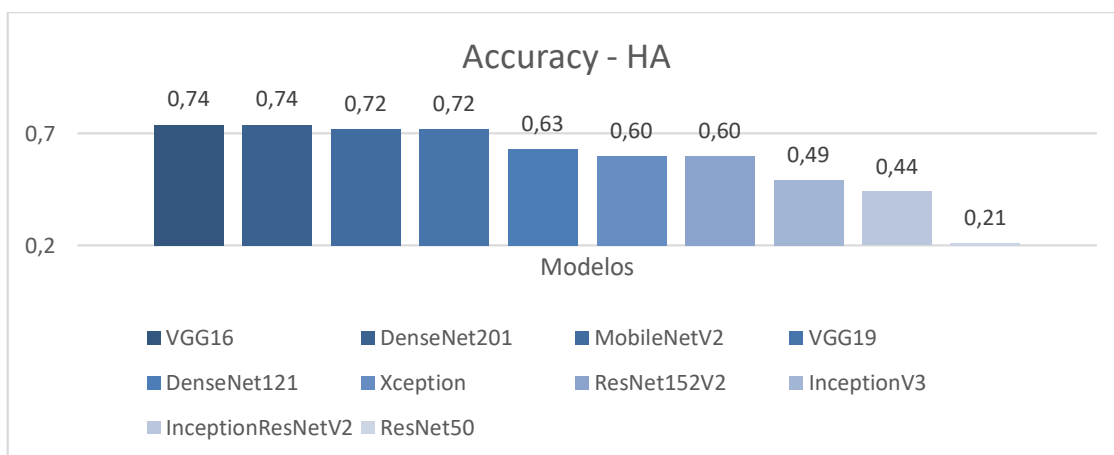


Figura 13. Accuracy HA

Es importante señalar que el objetivo en este trabajo es predecir el mayor número de imágenes en la clase correcta. En ese caso, conocer el resultado de la métrica Accuracy sería suficiente para resolver este problema. Sin embargo, en la Tabla 5 se puede evidenciar que los conjuntos de datos, MA y HA, están desbalanceados. Dada esa circunstancia, esta métrica tiende a ocultar errores de clasificación graves en las clases con pocas unidades, ya que esas clases son menos relevantes en comparación con las más grandes [63]. Por consiguiente, utilizar solamente esta métrica no ayuda a identificar el modelo más óptimo.

En ese sentido, en las Tablas 11 y 12 se exponen los resultados de las métricas de clasificación para cada uno de los datasets, MA y HA, que son: precisión, recall, y F1-Score. Cabe recalcar que precisión y recall actuarán como bloques de construcción para las fórmulas Balanced Accuracy y F1-Score que se mostrarán en las Figuras 9 y 10.

Tabla 11. Promedio de Métricas de Clasificación sobre el Conjunto de Datos MA

Modelo	Métricas de Clasificación - MA		
	Precisión (Avg)	Balance Accuracy	F1-Score (Avg)
MobileNetV2	0,87	0,85	0,86
VGG16	0,79	0,79	0,78
DenseNet121	0,83	0,76	0,77
InceptionResNetV2	0,72	0,73	0,71
DenseNet201	0,85	0,79	0,8
Xception	0,76	0,75	0,73
ResNet50	0,53	0,43	0,39
VGG19	0,85	0,85	0,84
InceptionV3	0,68	0,67	0,65
ResNet152V2	0,74	0,76	0,74

Tabla 12. Promedio de Métricas de Clasificación sobre el Conjunto de Datos HA

Modelo	Métricas de Clasificación - HA		
	Precisión (Avg)	Balance Accuracy	F1-Score (Avg)
MobileNetV2	0,68	0,69	0,68
VGG16	0,72	0,70	0,69
DenseNet121	0,64	0,63	0,62
InceptionResNetV2	0,51	0,44	0,43
DenseNet201	0,66	0,69	0,67
Xception	0,57	0,56	0,56
ResNet50	0,17	0,23	0,11
VGG19	0,74	0,7	0,68
InceptionV3	0,45	0,47	0,42
ResNet152V2	0,63	0,60	0,60

3.1.1. Precisión equilibrada en el conjunto de datos MA y HA

En las Tablas 11 y 12 se muestra la precisión equilibrada, en inglés conocida como accuracy balance, de cada modelo para cada conjunto de datos, MA y HA. Este indicador es el resultado de la media de los valores de recall de cada clase. Cabe recalcar que recall señala la probabilidad que existe en que una imagen de esa clase sea clasificada correctamente cuando el conjunto de datos está desequilibrado. Para ello, se asigna el mismo peso y la misma importancia a las clases minoritarias [63]. Por tanto, las clases más pequeñas acaban asumiendo una influencia más que proporcional en la fórmula, aunque su tamaño sea reducido en términos de número de unidades [64].

En ese sentido, la Figura 9 señala que el accuracy balanced de MobileNetV2 y VGG19 es igual a 85%, superando a los demás modelos. Por otro lado, la Figura 10 indica que VGG16

y VGG19 tienen un rendimiento igual al 70% en este indicador con respecto a los demás modelos empleados.

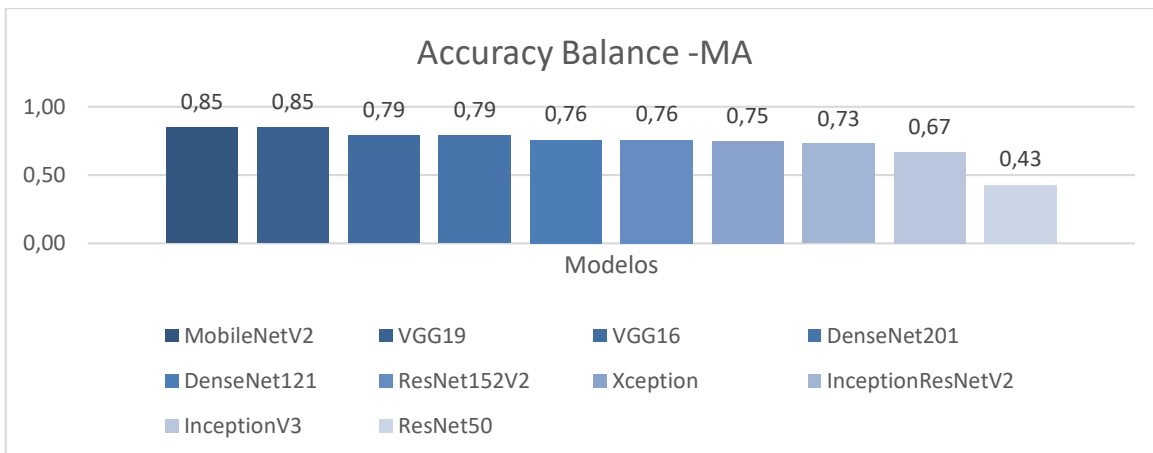


Figura 14. Accuracy Balance - MA

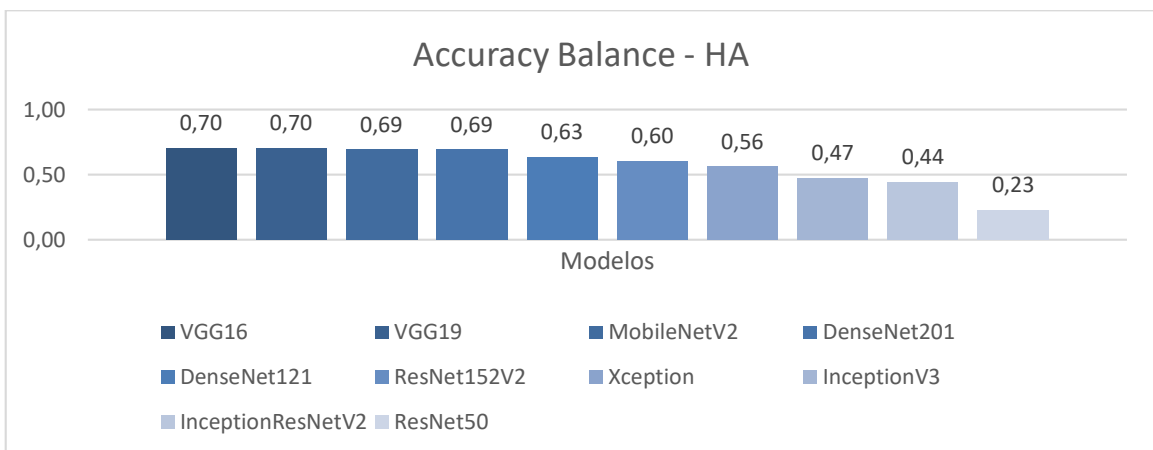


Figura 15. Accuracy Balanced – HA

3.1.2. F1-Score en el conjunto de datos MA y HA

En este experimento también se presenta los resultados del promedio de F1-Score por cada modelo. Esta métrica evalúa el rendimiento de cada modelo de clasificación a partir de la matriz de confusión. Además, agrega las medidas precisión y recall bajo el concepto de media armónica, la cual es útil para encontrar el mejor equilibrio entre las dos cantidades [65].

El resultado de las métricas de Precisión y Recall para cada conjunto de datos, se detalló en las Tablas 11 y 12. Sin embargo, para una mejor visualización de los resultados en las Figuras 11 y 12 se presenta una comparativa de los resultados alcanzados en la métrica F1-Score de cada modelo por conjunto de datos.

Específicamente, en la Figura 18, se puede evidenciar que MobileNetV2 es el modelo que alcanza una puntuación del 86%. Por otro lado, la Figura 19 indica que VGG16 logra el 69% en el conjunto de datos HA.

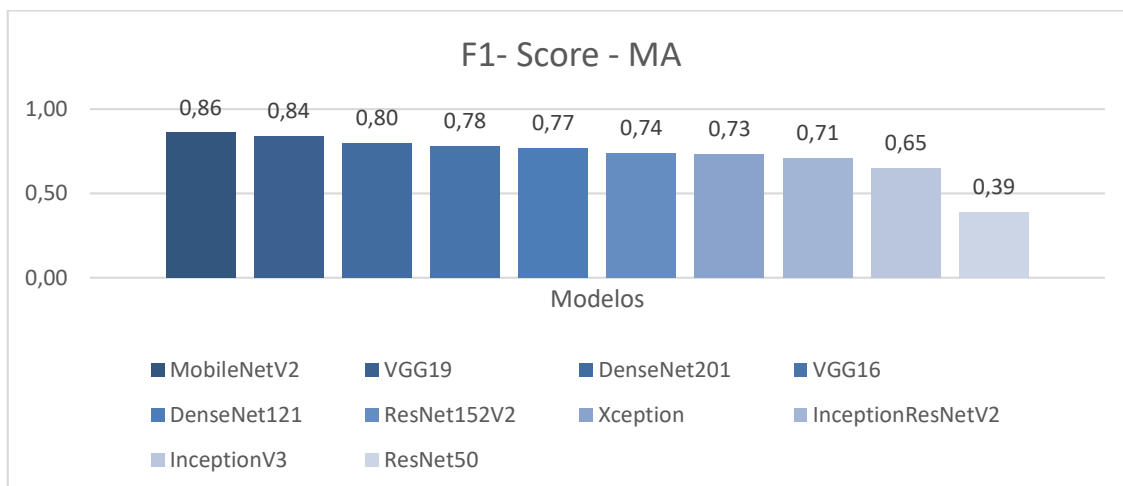


Figura 16. F1-Score - MA

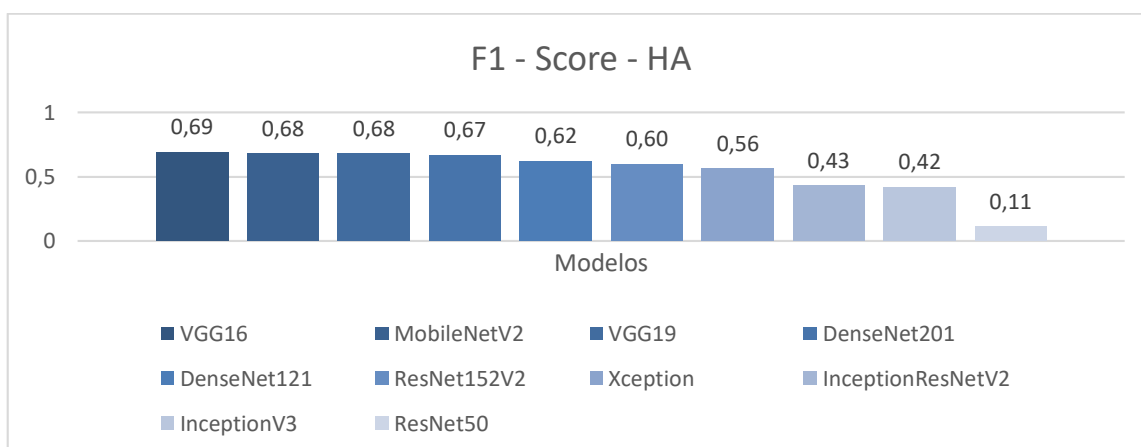


Figura 17. F1-Score - HA

Es importante notar que en las Figuras 18 y 19 se muestra el promedio de F1-score. Por tanto, un análisis más correcto de esta medida es estudiar su comportamiento clase por clase. En ese sentido, las Figuras 20 y 21 detallan los valores alcanzados en cada clase, pero sólo de los modelos que superan el 75% en MA y 65% en HA en dicho indicador.

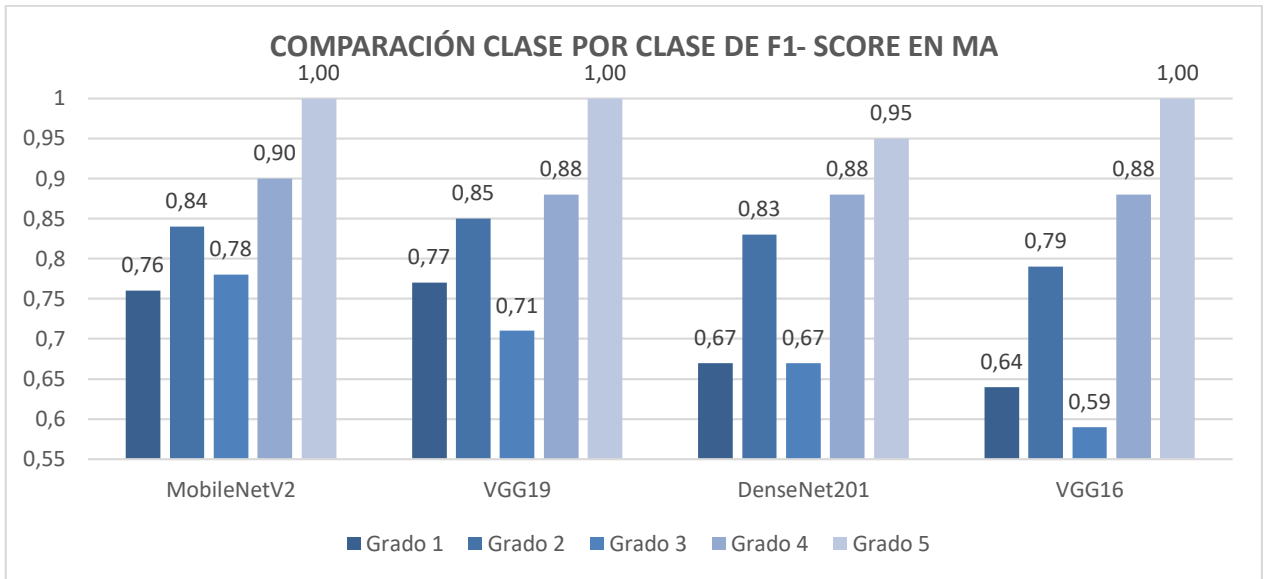


Figura 18. Comparación Clase por Clase F1-Score MA

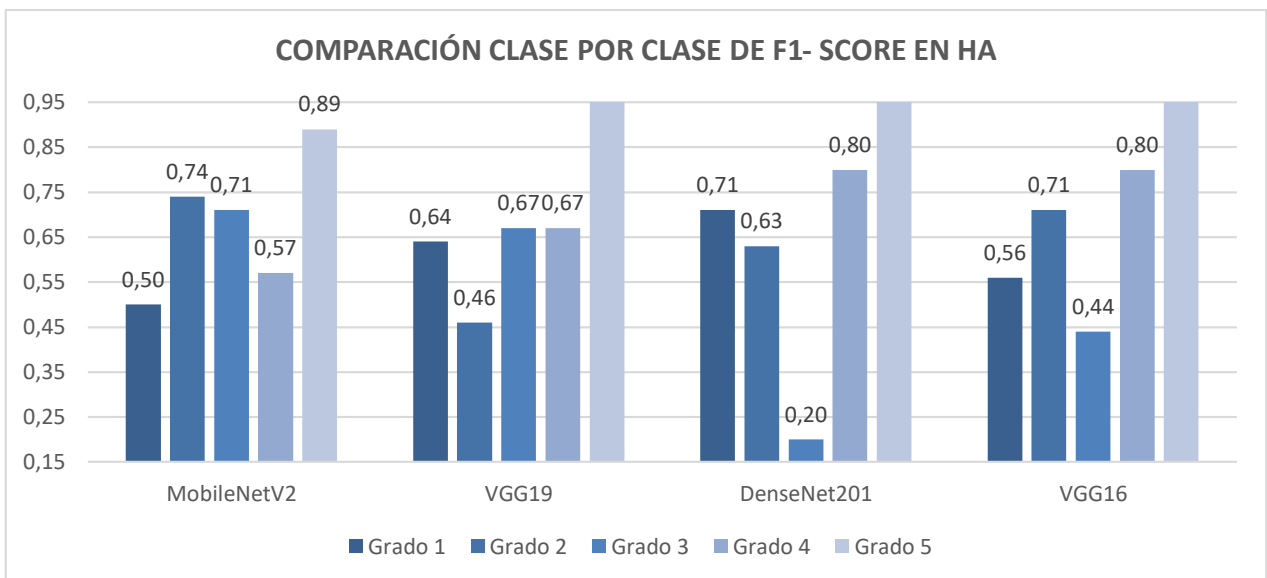


Figura 19. Comparación Clase por Clase F1-Score HA

A continuación, en la Figura 22 se muestra la comparativa de las métricas de clasificación entre los modelos que han alcanzado un valor mínimo de 82% y máximo de 86% en accuracy, de 79% a 85% en accuracy Balanced y de 80% a 86% en F1-Score sobre el conjunto de datos MA. Por otro lado, en la Figura 23 se muestra la comparativa de las métricas de clasificación entre los modelos que han alcanzado un valor entre 72 y 74% en accuracy, entre 69% y 70% en Accuracy balance y entre 67% y 69% en cualquiera de sus métricas sobre los conjuntos de datos HA, y se descarta, de los dos conjuntos de datos, los modelos cuyo accuracy sea menor al 50%.

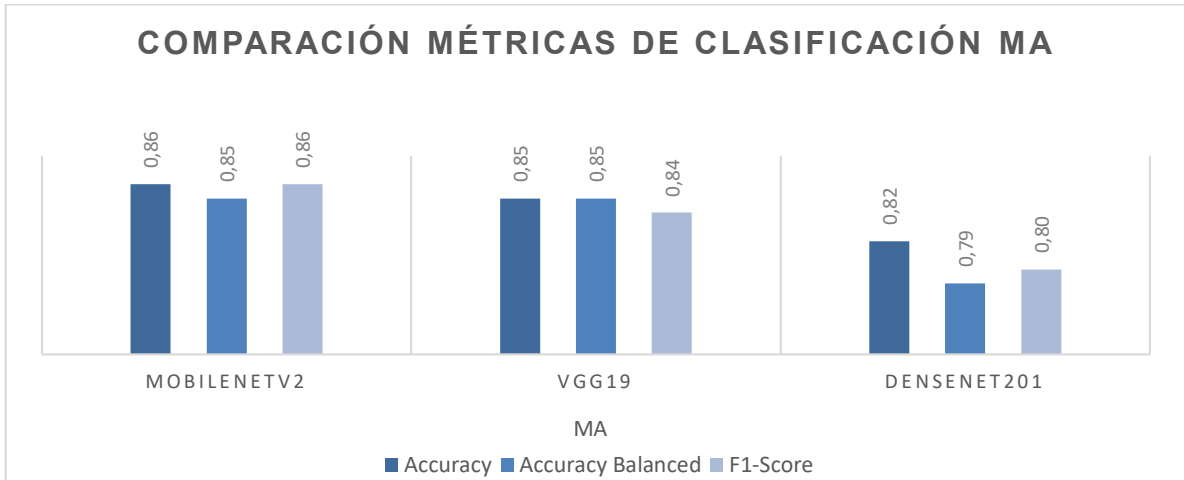


Figura 20. Comparativa Métricas de Clasificación MA

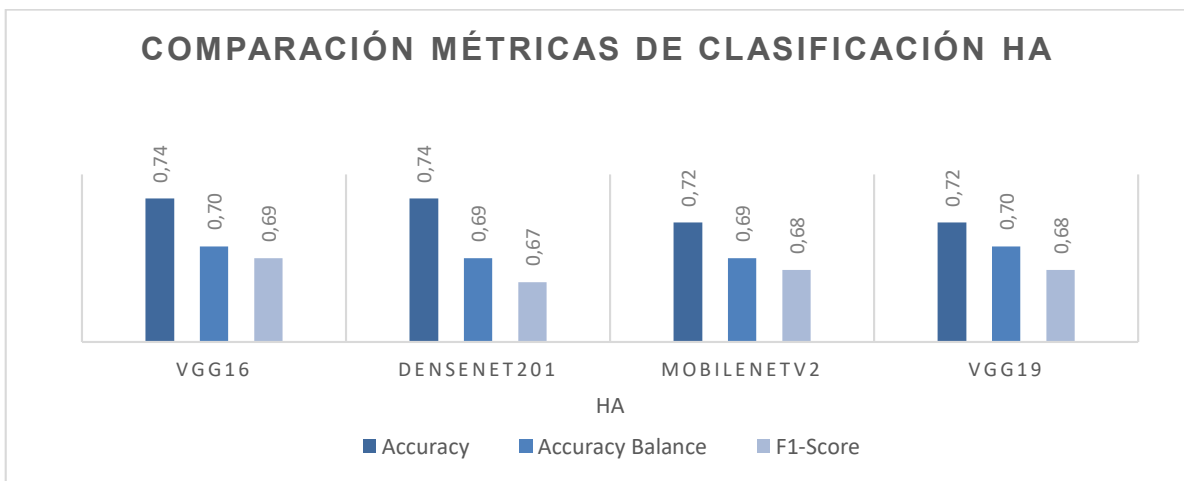


Figura 21. Comparativa Métricas de Clasificación HA

Sin embargo, analizando la métrica de F1-Score, clase por clase, de los modelos seleccionados en las Figuras 20 y 21 se puede notar que ciertas clases no están clasificadas correctamente, especialmente para el conjunto de datos HA, el cual contiene menos muestras que el conjunto de datos MA. Es por ello, que en la siguiente sección se muestra los resultados del segundo experimento realizado con aprendizaje por transferencia con aumento de datos.

3.2. Resultados de entrenamiento con aprendizaje por transferencia (con aumento de datos)

En este experimento se compara el rendimiento de las diez redes convolucionales pre-entrenadas con aprendizaje por transferencia usando un aumento de datos artificial con el paquete de Python, Augmentor, señalado en la sección 2.4.3, lo cual generó dos nuevos conjuntos de datos, MA₁ y HA₁, proporcionando un total de 1500 imágenes.

A continuación, las Tablas 13 y 14 muestran el comportamiento de la métrica de exactitud (accuracy) durante la etapa de entrenamiento con aumento de datos.

Tabla 13. Resultados de entrenamiento, validación y prueba MA₁

Modelo	Mampostería (MA ₁)			Épocas (/100)
	Exactitud			
	Entrenamiento	Validación	Prueba	
MobileNetV2	0,9958	0,8333	0,8667	12
VGG16	1,0000	0,8750	0,8467	49
DenseNet121	0,9979	0,8250	0,9067	22
InceptionResNetV2	0,9958	0,8167	0,8733	64
DenseNet201	0,9958	0,9000	0,9267	18
Xception	0,9958	0,8583	0,8467	25
ResNet50	0,5938	0,4500	0,4867	23
VGG19	0,9958	0,8667	0,8600	35
InceptionV3	0,9979	0,7583	0,8533	54
ResNet152V2	0,9979	0,8583	0,8600	13

Tabla 14. Resultados del entrenamiento, validación y prueba HA₁

Modelo	Hormigón Armado (HA ₁)			Épocas (/100)
	Exactitud			
	Entrenamiento	Validación	Prueba	
MobileNetV2	0,9979	0,925	0,9567	14
VGG16	0,9979	0,9333	0,9400	54
DenseNet121	0,9979	0,9167	0,9333	12
InceptionResNetV2	0,9958	0,8333	0,8667	53
DenseNet201	0,9979	0,9750	0,9400	17
Xception	0,9958	0,8917	0,9267	19
ResNet50	0,6458	0,3917	0,5067	17
VGG19	0,9979	0,9500	0,9400	51
InceptionV3	0,9958	0,8917	0,8800	33
ResNet152V2	0,9979	0,9250	0,9133	10

Se observa que la precisión de la validación mejora con el aumento de datos, con un +10%, a excepción de ResNet50. Por lo cual, ResNet50 fue descartado.

En las Figuras 24 y 25 se presenta los valores alcanzados de la métrica de exactitud (accuracy) por cada modelo sobre los dos conjuntos de datos, MA₁ y HA₁. En la Figura 24 se observa que DenseNet121 y DenseNet201 alcanzan un accuracy de 91% en comparación con los demás modelos. Asimismo, en la Figura 25, MobileNetV2 y DenseNet201 alcanzan un Accuracy del 95%.

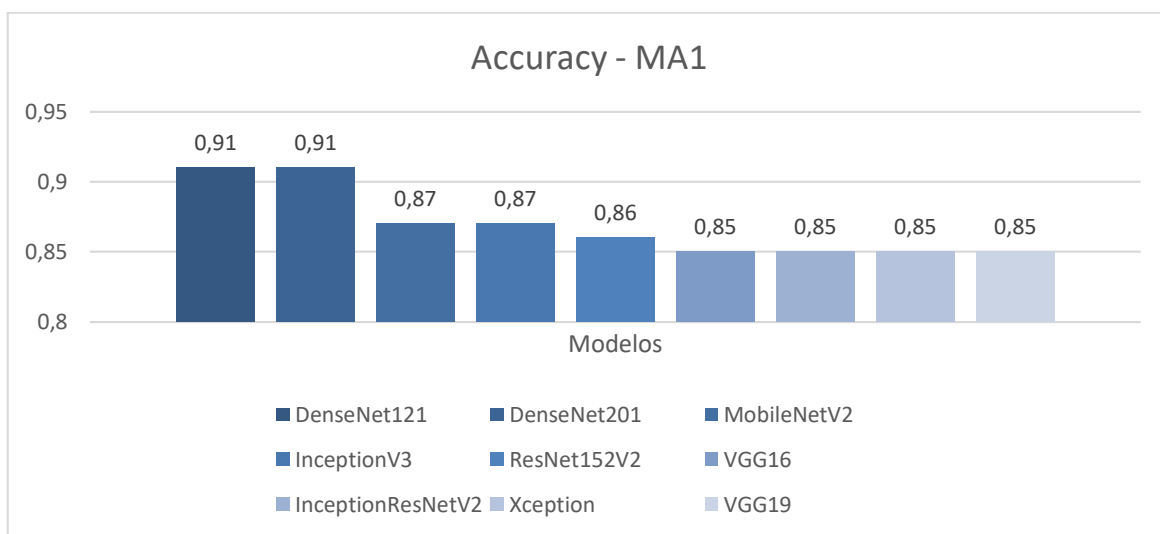


Figura 22. Accuracy MA₁

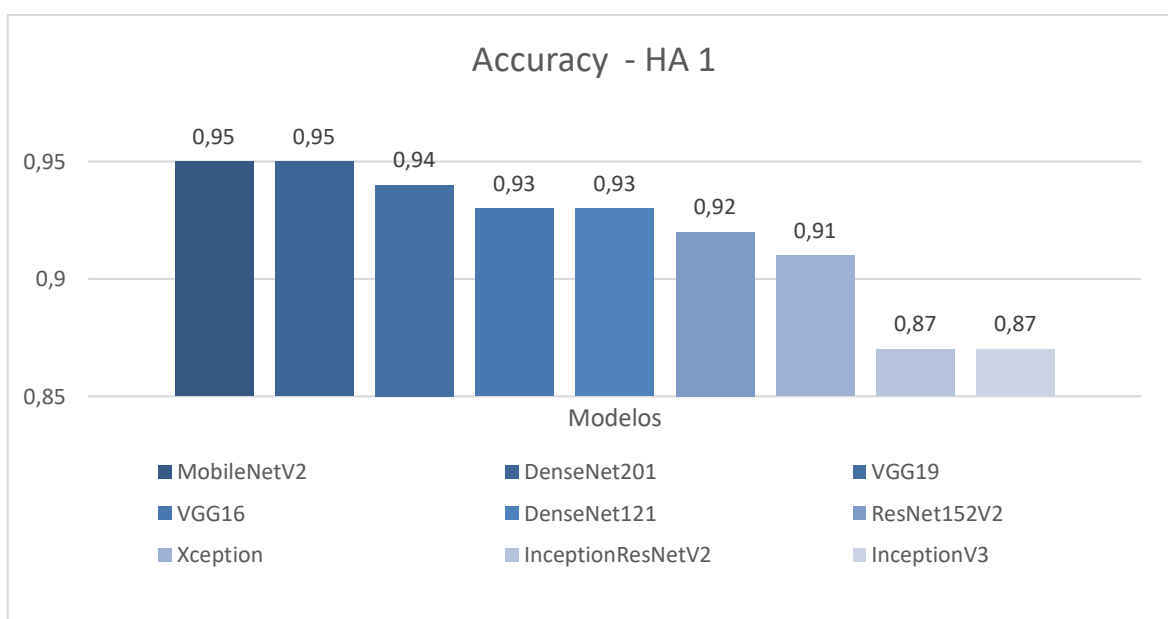


Figura 23. Accuracy - HA₁

3.2.1. Precisión equilibrada con aumento de datos MA₁ y HA₁

En la Figura 26, se observa que el mejor rendimiento sobre el conjunto de datos MA₁ se obtiene con DenseNet201, cuya precisión es igual al 91%. Asimismo, es el caso de DenseNet121 que es igual al 90% y MobileNetV2 tiene un 87% en la métrica de precisión; por tanto, se consideran como buenos predictores.

Por otro lado, en la Figura 27 se evidencia que el mejor rendimiento se obtiene con MobileNetV2 y DenseNet201, cuya precisión es igual al 95%. Aunque en este caso, la mayoría de modelos superan una precisión del 90%.

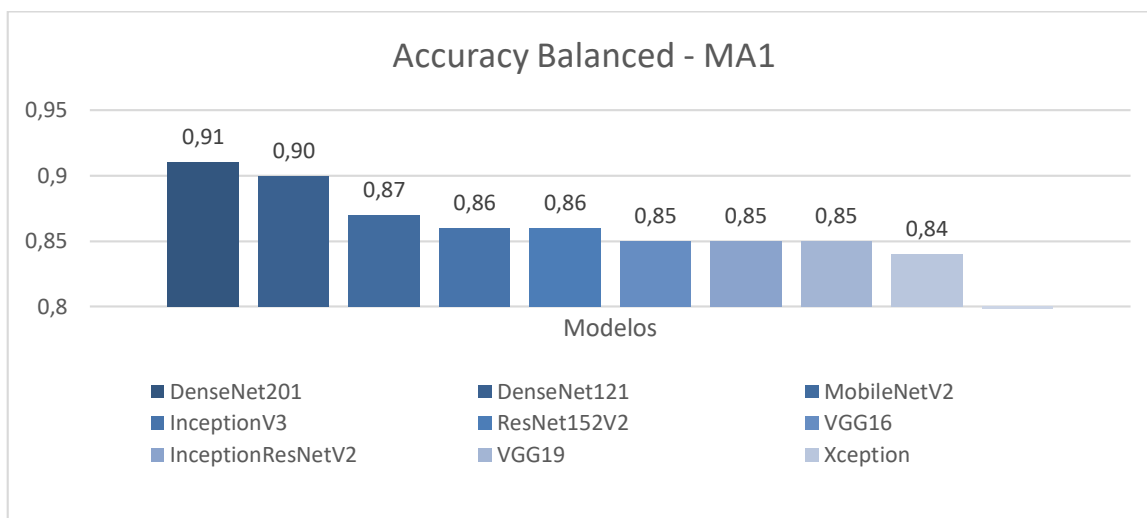


Figura 24. Accuracy Balanced MA₁

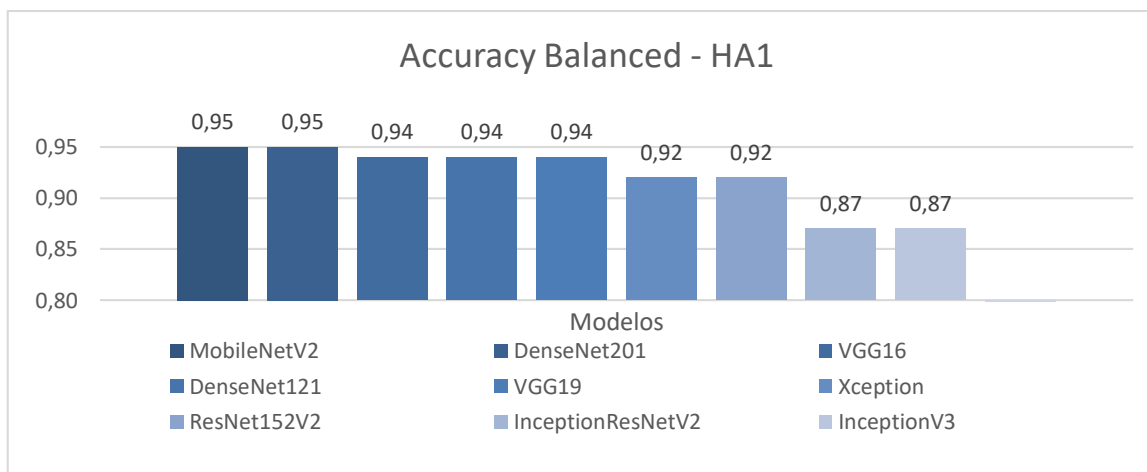


Figura 25. Accuracy Balance HA₁

3.2.2. F1-Score con aumento de datos MA₁ y HA₁

La Figura 28 señala que DenseNet201 es el modelo que alcanza el mejor rendimiento en comparación con los demás modelos, alcanzando el 91%. Mientras que, en la Figura 29, DenseNet201 y MobileNetV2 son los modelos que alcanzan el 95%.

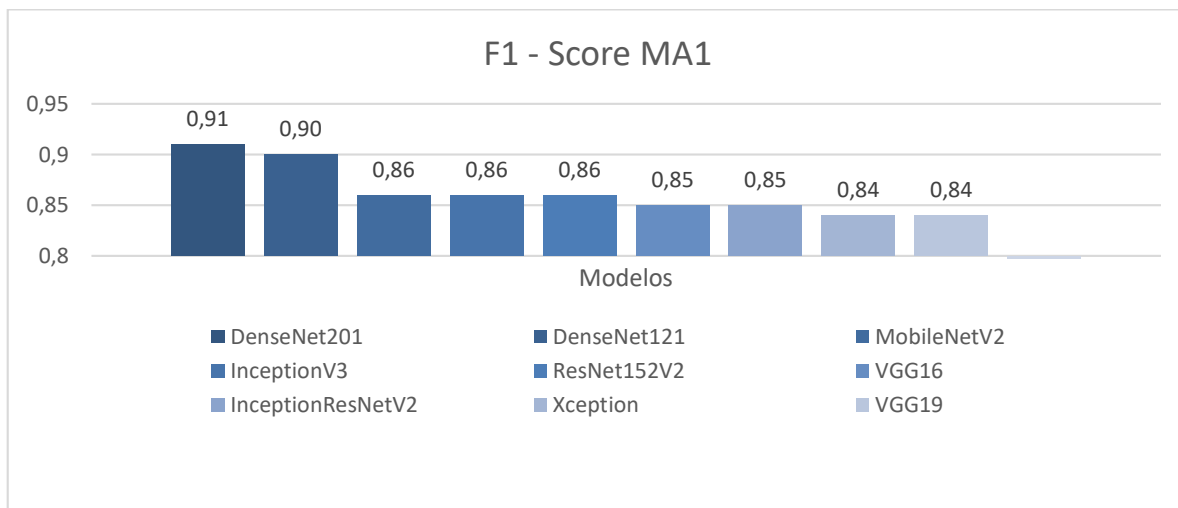


Figura 26. F1-Score MA1

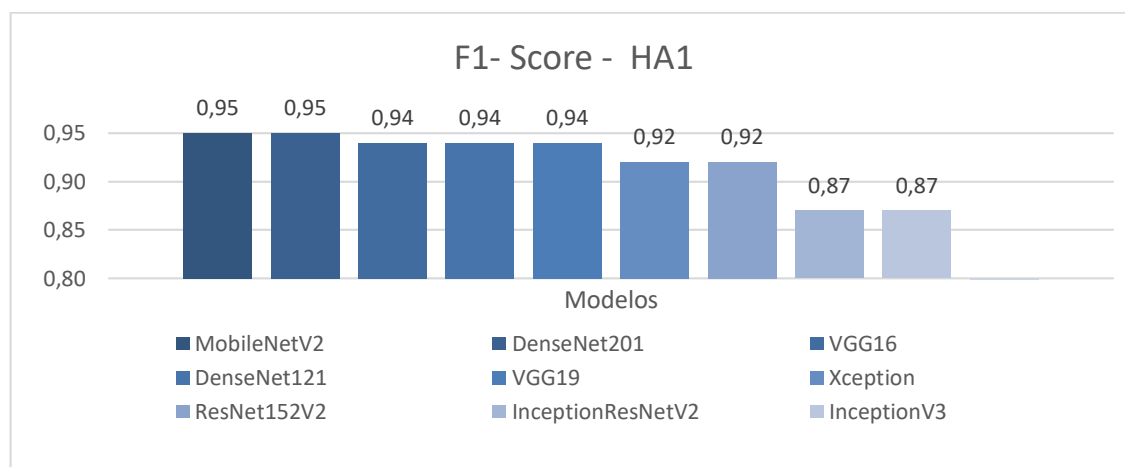


Figura 27. F1 - Score HA1

De la misma forma, en las Figuras 30 y 31 se muestra una comparativa entre las métricas de clasificación de los modelos que alcanzaron un rendimiento superior al 80%. En ese caso, la Figura 30 evidencia que DenseNet201 es el modelo que mejor se comporta para este conjunto de datos. Por otra parte, en la Figura 31 se señala que Dense201 y MobileNeV2, son los modelos que alcanzan, en sus métricas, un valor del 95%.

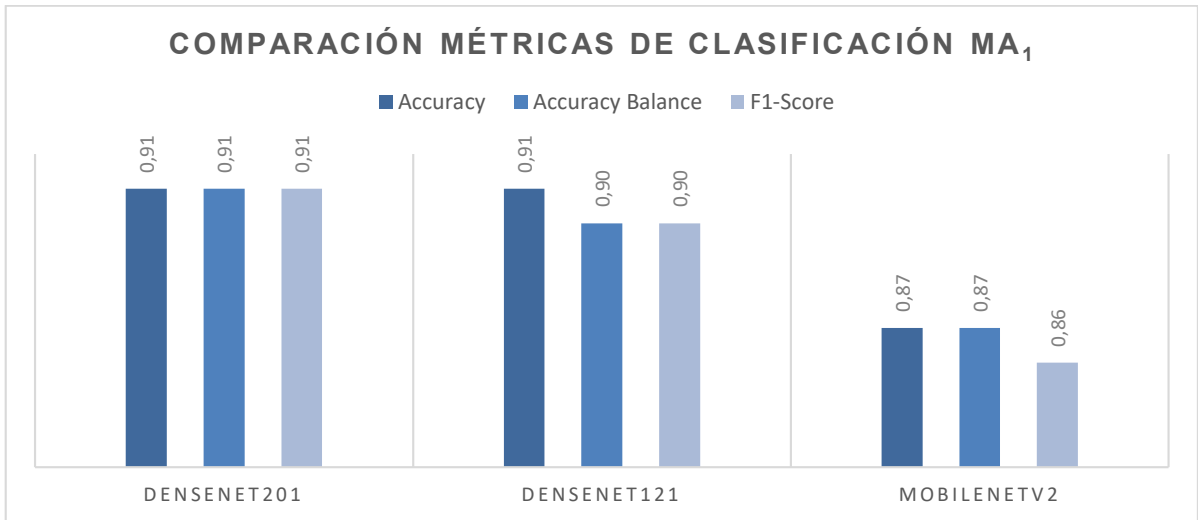


Figura 28. Comparativa Métricas de Clasificación MA1

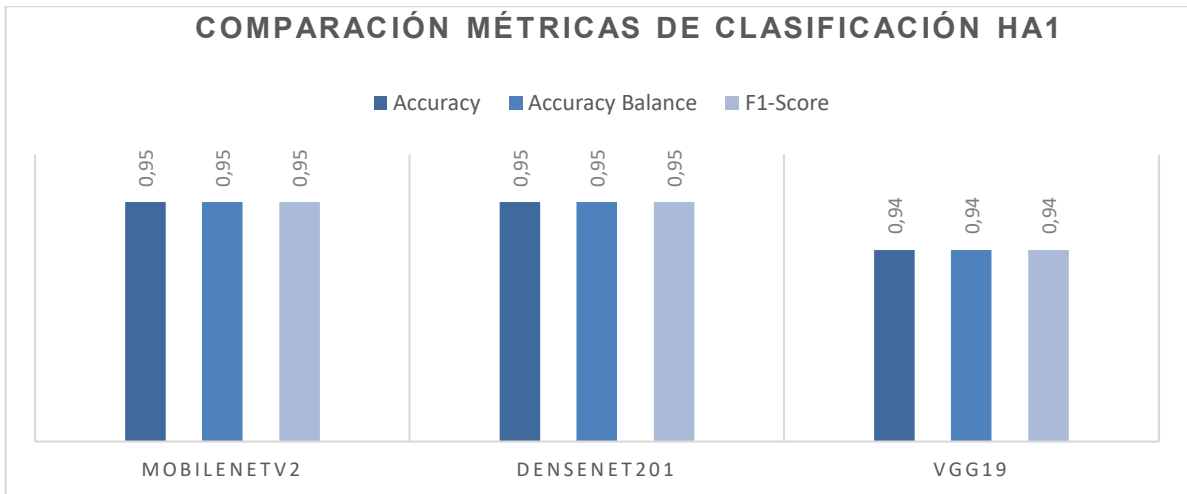


Figura 29. Comparativa Métricas de Clasificación HA1

3.2.3. F1-Score con aumento de datos clase por clase MA₁ y HA₁

Nuevamente, es importante notar que en las figuras anteriores se muestra el promedio de las métricas. Sin embargo, para un análisis más correcto de los resultados, el siguiente paso es mostrar los valores de F1-Score por cada clase.

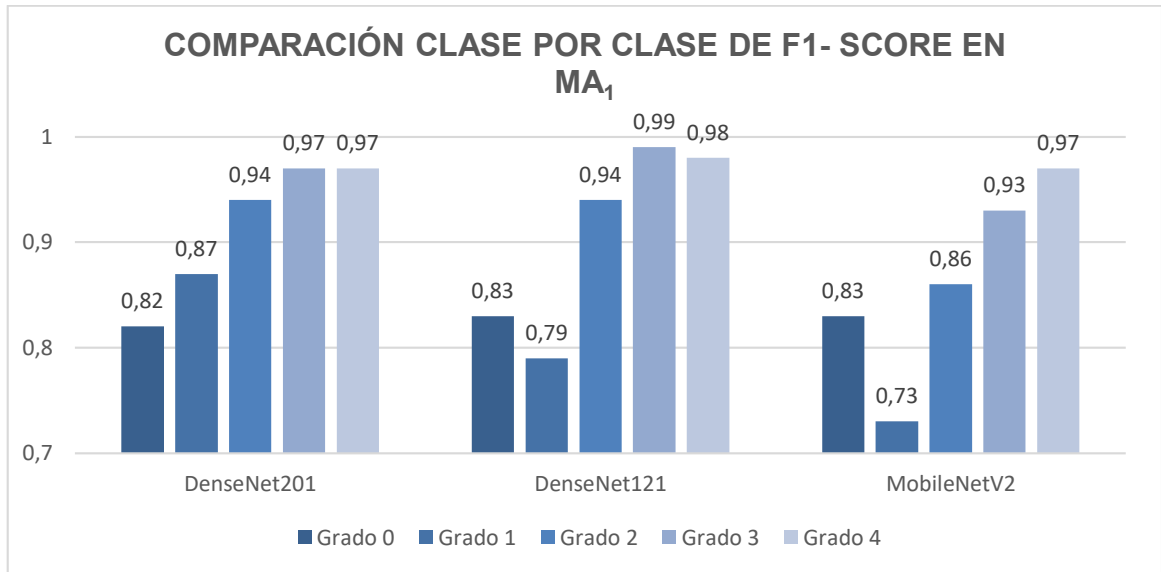


Figura 30. Comparativa Clase por Clase de F1-Score en MA_1

Considerando los resultados de la Figura 32 se puede evidenciar que DenseNet201, es el modelo que clasifica de mejor manera las imágenes en cada grado, ya que en las 5 clases el indicador de F1-Score supera el 80% en comparación con los demás modelos.

Por otro lado, en la Figura 33 se evidencia que el indicador F1-Score en los tres modelos, MobileNetV2, DenseNet201 y VGG19, superan el 90%, por tanto, los tres modelos clasifican correctamente las imágenes en cada grado.

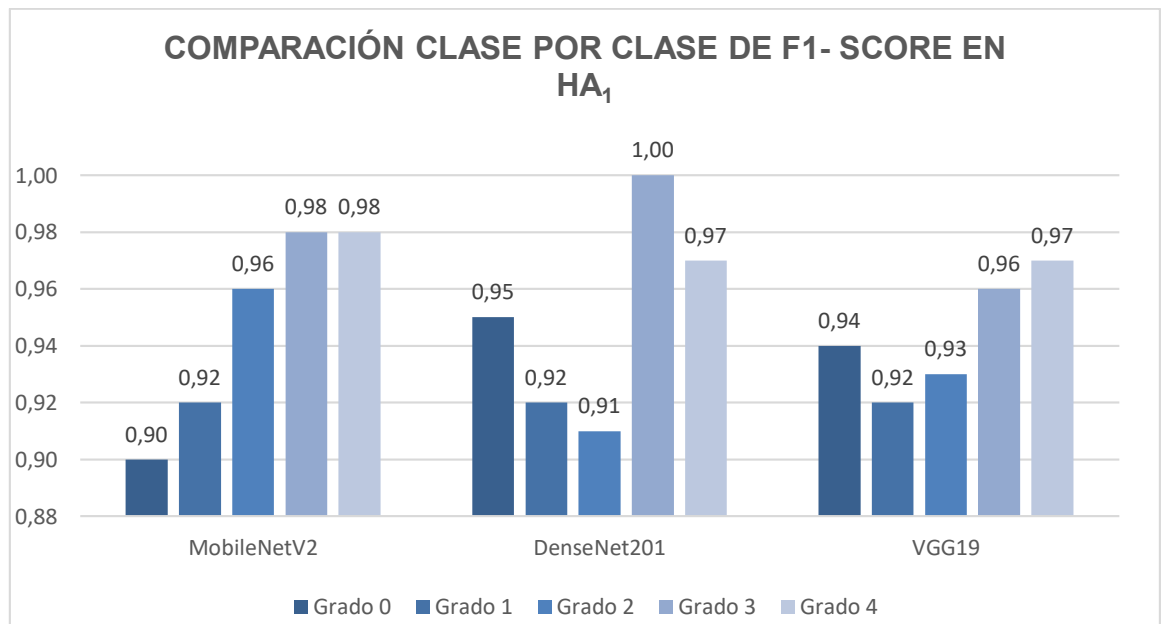


Figura 31. Comparativa Clase por Clase de F1-Score en HA_1

Luego de detallar los resultados de cada métrica por cada grado de daño y tipo de estructura, es conveniente resumir dichos valores, para así tomar una decisión del modelo o modelos más óptimos. Dicho esto, en las Figuras 34 y 35 se muestran una comparativa del rendimiento alcanzado durante los dos experimentos realizados en base a los valores resumidos en las Tablas 15 y 16.

Tabla 15. Métricas de Evaluación para el Conjunto de Datos MA y MA1

Técnica	Modelos - Conjunto de Datos	Métricas de Clasificación			
		Accuracy	Precision	Accuracy Balanced	F1-Score
Con aumento de datos	DenseNet121 MA1	0,91	0,91	0,90	0,90
	DenseNet201 MA1	0,91	0,92	0,91	0,91
	MobileNetV2 MA1	0,87	0,87	0,87	0,86
	VGG16 MA1	0,85	0,85	0,85	0,85
Sin aumento de datos	MobileNetV2 MA	0,86	0,87	0,85	0,86
	DenseNet201 MA	0,82	0,85	0,79	0,80
	VGG16 MA	0,80	0,79	0,79	0,78
	DenseNet121 MA	0,80	0,83	0,76	0,77

Tabla 16. Métricas de Evaluación para el Conjunto de Datos HA y HA1

Técnica	Modelos- Conjunto de datos	Métricas de Clasificación			
		Accuracy	Precision	Accuracy Balanced	F1-Score
Con aumento de datos	MobileNetV2 HA1	0,95	0,95	0,95	0,95
	DenseNet201 HA1	0,95	0,95	0,95	0,95
	VGG19 HA1	0,94	0,94	0,94	0,94
	VGG16 HA1	0,93	0,94	0,94	0,94
	DenseNet121HA1	0,93	0,94	0,94	0,94
Sin aumento de datos	VGG16 HA	0,74	0,72	0,70	0,69
	DenseNet201 HA	0,74	0,66	0,69	0,67
	MobileNetV2 HA	0,72	0,68	0,69	0,68
	VGG19 HA	0,72	0,74	0,70	0,68
	DenseNet121 HA	0,63	0,64	0,63	0,62

En la Figura 34, se evidencia que el modelo DenseNet201, en el conjunto de datos MA1, muestra los mejores resultados en las cuatro métricas que se analizan en este trabajo, y el modelo con menor rendimiento es DenseNet121 en el conjunto de datos MA.

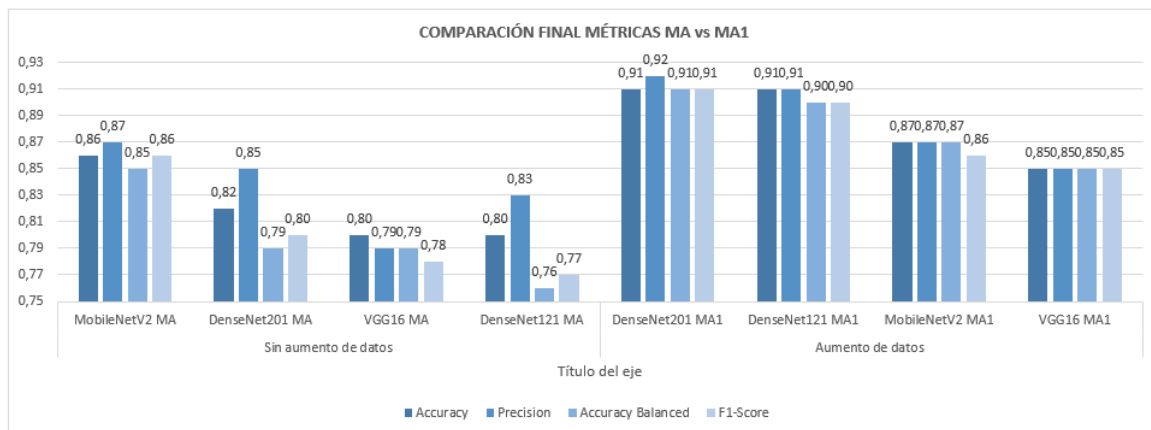


Figura 32. Comparación Resultados Final Métricas MA vs MA1

En la Figura 35, se evidencia que el modelo DenseNet201, en el conjunto de datos HA1, muestra los mejores resultados en las cuatro métricas que se analizan en este trabajo, y el modelo con menor rendimiento es DenseNet121 en el conjunto de datos HA.

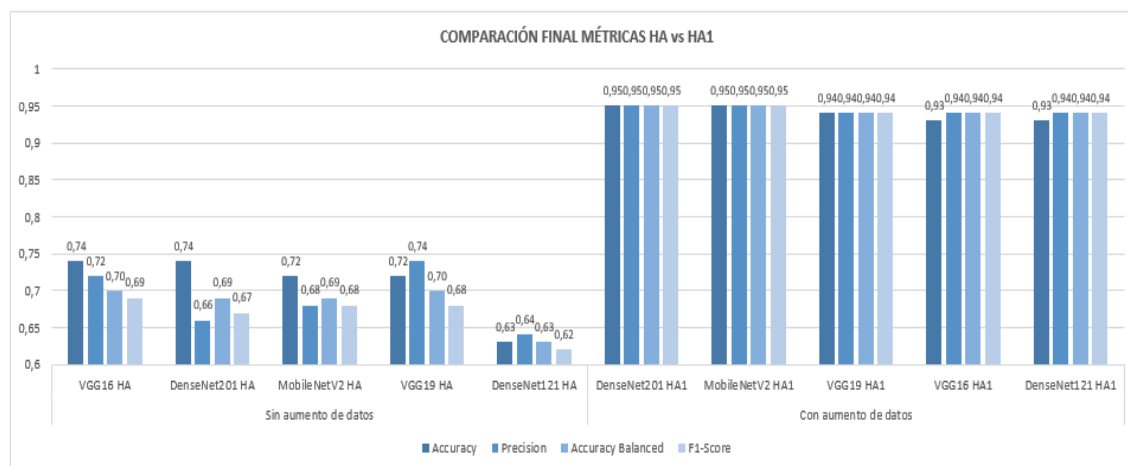


Figura 33. Comparación Resultados Final Métricas HA vs HA1

Finalmente, en la Figura 34 se muestra el diagrama de flujo para construir un modelo de aprendizaje profundo, para esto, se debe construir u obtener el conjunto de datos, luego dichos datos deben ser procesados. Durante el procesamiento se aplica la conversión a color RGB, la estandarización del tamaño y se define el formato de las imágenes, además, se aplica el suavizado de imágenes para eliminar el ruido y finalmente, se aplica el aumento de datos.

Esta técnica se aplica cuando los datos son escasos. Posterior a esto, se realiza la división del dataset, en entrenamiento, validación y prueba. En la fase de modelado es importante seleccionar el modelo de aprendizaje. En este caso se usó las CNN debido a que en este proyecto se emplean fotografías digitales. Además, en esta etapa, se define la función de pérdida del modelo, las métricas de evaluación, y los hiperparámetros. Luego de dicha etapa, se procede a entrenar el modelo y se evalúa mediante las métricas de exactitud, precisión, y f1-score. Una vez que se obtenga el mejor rendimiento del modelo, se procede a la implementación del modelo en la organización, para esto se ingresan nuevos datos los cuales serán clasificados de acuerdo a las etiquetas respectivas.

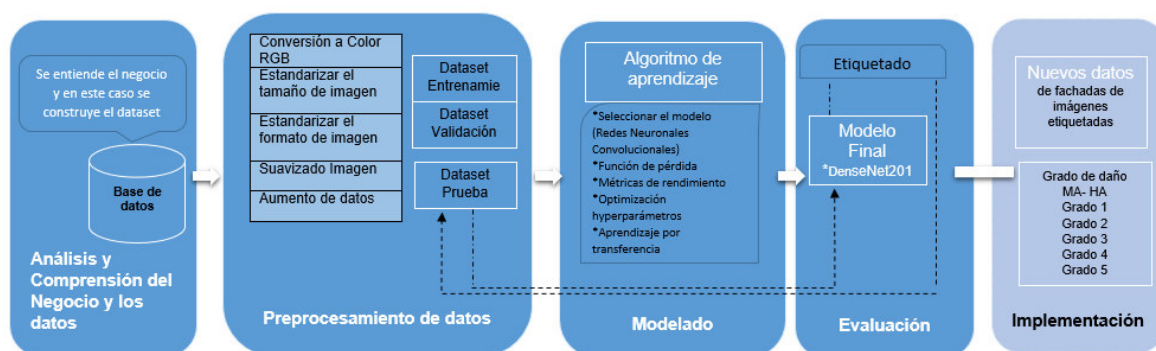


Figura 34. Diagrama de Flujo para Construir un Modelo de Aprendizaje Autónomo.

3.3. Discusión

Se han realizado dos experimentos para cumplir con el objetivo relacionado con la propuesta de un modelo de clasificación de imágenes de estructuras de edificios dañados por un sismo. Si se compara los resultados de las Figuras 32 y 33, se refleja que DenseNet201 es el modelo que presenta mejores resultados que los otros modelos seleccionados para los conjuntos de datos MA_1 y HA_1 . Aunque, los demás modelos como MobileNetV2, VGG16, VGG19, DenseNet121, no decepcionan, ya que los resultados de estos superan el 85% en accuracy, precision, accuracy balanced y F1-Score.

Cabe recalcar que MA_1 y HA_1 son producto de aplicar la técnica de aumento de datos en MA y HA. Esta técnica fue necesaria aplicar debido a que en el primer experimento (sin aumento de datos) los valores de las métricas por cada clase (grado de daño) resultaron bajos (<0.70), especialmente en las clases con pocas muestras, tal como se muestra en las Figuras 20 y 21.

En ese sentido, la clave para conseguir que una CNN funcione con precisión es entrenarla con una gran variación de muestras que puedan representar la distribución de cada clase.

Sin embargo, en cualquier ámbito, social, económico o médico, la tarea de obtener muestras no es una tarea fácil, y más aún, cuando el conjunto de datos no es proporcionado por el negocio. De hecho, en este trabajo de titulación se construyó los conjuntos de datos, MA y HA.

Por otro lado, se podría cuestionar por qué no se creó un conjunto de datos con ambas bases de datos y utilizar el aprendizaje de transferencia para entrenar los modelos. No obstante, esta opción fue considerada, pero cuando se experimentó el entrenamiento de las dos bases de datos juntas no se obtuvo, en ninguna de sus métricas, buenos resultados, esto se dio principalmente por el desbalance y escases de muestras que existe entre los dos conjuntos de datos, MA y HA.

4. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

Sin duda alguna, la etapa principal del proyecto que hace referencia a la obtención del conjunto de datos y el preprocesamiento de los datos es de gran importancia y determinante para el éxito o fracaso de cualquier proyecto. Se dice que la captación de la imagen es el 50% del sistema y sería un error muy serio pensar que se puede compensar una imagen inadecuada con algún algoritmo; es mejor tener una buena calidad de imagen y utilizar menos algoritmos para su mejoramiento. Es por lo que se empleó gran cantidad del tiempo de desarrollo del proyecto en esas dos fases, ya que inicialmente se tenía la idea de que las redes convolucionales iban hacer toda la magia, sin embargo, con el transcurso del tiempo y tras varios experimentos, se pudo notar que no todo se basa en la herramienta empleada, sino como se ha mencionado ya, todo depende de la calidad de los datos.

Dado que las imágenes se descargaron de diferentes fuentes, las cuales nunca tuvieron el objetivo de ser empleadas en este tipo de proyecto, las fotografías fueron tomadas en diferentes ángulos, con personas cruzando o personal de emergencia, como bomberos o policías interviniendo durante los hechos; incluso hay presencia de animales, lo cual dificultó el proceso de clasificación manual. Sin embargo, se trató de focalizar en mayor medida y dentro de lo posible en el daño y rescatar las imágenes candidatas que podrían alimentar al modelo.

Además, existe el paradigma, para las redes neuronales, de la necesidad de un gran volumen de datos para realizar un buen entrenamiento. En este caso, el conjunto de datos construido tiene un total de 605 imágenes, un número relativamente pequeño; sin embargo, con este número de datos combinado con la técnica de aprendizaje por transferencia se pudo lograr resultados con una exactitud aceptable dada por el modelo DenseNet201 con un 91% en MA₁ y 95% en HA₁. No obstante, MobileNetV2 y VGG16, son los modelos que presentan 86% y 74% en el indicador de accuracy sobre los conjuntos de datos MA y HA, respectivamente. Por tanto, son los modelos que mejor rendimiento muestran con respecto a los demás.

La eficacia global de los modelos se juzgó utilizando las métricas de evaluación de exactitud, precisión, recall y puntuación f1, calculadas a partir de las matrices de confusión. En general se puede concluir que las CNN se pueden emplear en la clasificación de imágenes, pero el éxito o fracaso de este tipo de proyecto depende de las dos primeras etapas de CRISP-DM, que son la recolección de datos y el preprocesamiento de los mismos.

4.2. RECOMENDACIONES

En este trabajo la tarea de etiquetado fue bajo un criterio teórico de la EMS. Sin embargo, conocer los diferentes tipos de estructuras y los grados de daño de la EMS requieren, no solo del conocimiento de esas áreas, sino de la experiencia que se adquiere al trabajar en este ámbito. Por tanto, se sugiere que el etiquetado de las imágenes sea revisadas por un equipo de especialistas en esta área.

Asimismo, se recomienda iniciar un proyecto para el levantamiento de información (imágenes candidatas) después de cada terremoto, lo cual ayude a aumentar la cantidad y la calidad de datos. En referencia a esta última, se puede decir que pesa más la calidad que la cantidad. Por tanto, es necesario que las imágenes sean procesadas previo a ingresar a cualquier modelo. En este trabajo, se estandarizó el tamaño de las imágenes de acuerdo a los CNN a usar, debido a que estas exigen el ingreso de imágenes con un tamaño mínimo. También, el formato de cada imagen se debe establecer acorde a la capacidad computacional que disponga el proyecto. Además, fijar el número de canales, eliminar ruido y suavizar las imágenes son pasos que se recomienda considerar para obtener mejores resultados.

Por otro lado, para lograr disminuir drásticamente el tiempo de entrenamiento y más aún cuando no se dispone de recursos computacionales, Google Colaboratory es una herramienta bastante útil, ya que esta ofrece uno de los dispositivos más potentes y existentes en el procesamiento gráfico, conocido como GPU.

El tener la predicción del modelo de la clasificación de imágenes por grado de daño según el tipo de estructura hormigón armado y mampostería, permitirá tener una descripción cualitativa y más objetiva de los efectos de los sismos sobre las infraestructuras.

REFERENCIAS BIBLIOGRÁFICAS

- [1] C. Bravo Cobeña, P. Valdivieso Guerra y R. Arregui Pozo, «Los Sistemas de Información en la Toma de Decisiones Gerenciales en las Empresas Comerciales de Portoviejo,» *Revista ECA Sinergia*, vol. 9, nº 2, pp. 45-54, 2018.
- [2] A. Muñoz Cañavate, «Sistemas de Información en las Empresas,» 2003.
- [3] I. M. Shaluf, «Tipos de desastres en Malasia: una descripción general,» *Prevención y gestión de desastres: una revista internacional*, vol. 16, nº 5, pp. 704-717, 2007.
- [4] N. I. o. B. Sciences, «Earthquake -Resistant Design Concepts,» Building Seismic Safety Council, Washintong, DC, 2010.
- [5] C. S. Europea, «Escala Macrosísmica Europea,» vol. 2017, 2008.
- [6] G. Grüntal y F. Lorenzo Martín, «Escala Macrosísmica Europea 1998,» G.Grüntal, Luxembourg, 2009.
- [7] I. EPN, «Informe Sísmico Especial Nro.18-2016,» 4 mayo 2016. [En línea]. Available: <http://www.igepn.edu.ec/1324-informe-sismico-especial-n-18-2016.html>.
- [8] R. Bonett, *Vulnerabilidad y riesgo sísmico de edificios. Aplicacion a entornos moderados en zonas de amenaza alta y moderada*, Barcelona, 2003.
- [9] S. K. Chinnamgari, «R. Machine Learning Projects: Implement supervised, unsupervised and reinforcement learning using R,» Packt publishing, 2019, pp. 13-23.
- [10] R. Valencia, «Aprendizaje Keras,» Free unaffiliated eBook created from Stack Overflow Contributors, 2019.
- [11] T. Ayodele, «Types of machine learning algorithms,» *New advances in machine learning*, vol. 3, pp. 19-48, 2010.
- [12] R. Sutton y A. Barto, *Reinforcement Learning, second edition: An Introduction*, MIT press, 2018.
- [13] F. Chollet, *Deep Learning with Python*, Shelter Island, NY: Maning Publications Co., 2018.
- [14] J. J. Beunza Nuin, E. Puertas Sanz y E. Condés Moreno, *Manual Práctico de Inteligencia Artificial en Entornos Sanitarios*, Barcelona: Elvesier España, S.L.U, 2020.

- [15] P. Larrañaga, I. Inza y A. Moujahid, «Tema 8: Redes Neuronales,» Universidad del País Vasco.
- [16] R. Yamashita, M. Nishio, K. Togashi y R. K. Gian Do, «Convolutional neural networks: an overview and application in radiology,» *Insights into imaging*, vol. 9, nº 4, pp. 611- 629, 2018.
- [17] G. Viera-Maza, *Procesamiento de imágenes usando OpenCV aplicado en Raspberry Pi para la clasificación del cacao*, Piura, 2017.
- [18] C. Boncelet, «Image noise models,» Academic Press, 2009, pp. 143-168.
- [19] P. Patidar, M. Gupta, S. Srivastava y A. K. Nagawat, «Image De-noising by Various Filters for Different Noise,» *International journal of computer applications*, vol. 9, nº 4, pp. 45-50, 2010.
- [20] M. Jaramillo, A. Fernández y E. Salazar, «Filtro de Mediana sobre Redes Neuronales,» 2016.
- [21] D. C. Ciresan, «Flexible, high performance convolutional neural networks for image classification,» de *Twenty-second international joint conference on artificial intelligence*, 2011.
- [22] A. Arista Jalife, G. Calderon Auza, A. Fierro Radilla y M. Nakano, «Clasificación de Imágenes Urbanas Aéreas: Comparación entre Descriptores de Bajo Nivel y Aprendizaje Profundo,» *Información tecnológica*, vol. 28, nº 3, pp. 209-224, 2017.
- [23] W. Li, H. Liu, Y. Wang, Z. Li, Y. Jia y G. Gui, *IEEE*, vol. 7, pp. 36274--36284, 2019.
- [24] «Deep learning for biological image classification,» *Expert Systems with Applications*, vol. 85, pp. 114-122, 2017.
- [25] J.-K. Park, B.-K. Kwon, J.-H. Park y D.-J. Kang, «Machine learning-based imaging system for surface defect inspection,» *International Journal of Precision Engineering and Manufacturing-Green Technology*, vol. 3, nº 3, pp. 303-310, 2016.
- [26] R. Wirth y J. Hipp, «CRISP-DM: hacia un modelo de proceso estándar para la minería de datos,» de *En Actas de la 4ª conferencia internacional sobre las aplicaciones prácticas de descubrimiento de*, Londres, 2000.
- [27] J. L. Abreu, «Abreu, J. (2012). Hipótesis, método & diseño de investigación (hypothesis, method & research design),» *Daena: International Journal of Good Conscience*, vol. 7, nº 2, pp. 187-197, 2012.

- [28] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer y R. Wirth, «CRISP-DM 1.0: Step-by-step data mining guide,» Germany, 2000.
- [29] S. Moro, R. Laureano y P. Cortez, «Uso de la Minería de Datos para el Marketing Directo Bancario: Una aplicación del CRISP-DM,» 2011.
- [30] R. Borja-Robalino, A. Monleón-Getino y J. Rodellar, «Estandarización de métricas de rendimiento para clasificadores Machine y Deep Learning,» *Revista Ibérica de Sistemas y Tecnologías de la Información*, n° 30, pp. 172-184, 2020.
- [31] C. Beauval, H. Yepes, P. Palacios, M. Segovia, A. Alvarado, Y. Font, J. Aguilar, L. Troncoso y S. Vaca, «An Earthquake Catalog for Seismic Hazard Assessment in Ecuador,» *Bulletin of the Seismological Society of America*, vol. 2A, n° 103, pp. 773-786, 2013.
- [32] IG-EPN, «Catalogos-sísmicos,» [En línea]. Available: <https://www.igepn.edu.ec/catalogos-sismicos>. [Último acceso: 17 diciembre 2021].
- [33] F. Rivadeneira, M. Segovia, A. Alvarado, J. Egred, L. Troncoso, S. Vaca y H. Yepes, «Breves fundamentos sobre los terremotos en el Ecuador,» Instituto Geofísico De la Escuela Politécnica Nacional, Quito, 2007.
- [34] T. Moncayo, «Terremotos mayores a 6.5 en escala Richter ocurridos en Ecuador desde 1900 hasta 1970.,» vol. 21, n° 2, pp. 55-64, 2017.
- [35] J. A. Taquía Gutiérrez, «El Procesamiento de Imágenes y su Potencial Aplicación en Empresas con Estrategia Digital,» Lima, Perú, 2017.
- [36] K. Applications, «Keras Applications,» [En línea]. Available: <https://keras.io/api/applications/#usage-examples-for-image-classification-models>. [Último acceso: 10 09 2021].
- [37] C. Aggarwal C, *Neural Networks and Deep Learning*, NY, USA: Yorktown Heights, 2018.
- [38] M. Bloice, C. Stoker y A. Holzinger, «Agumentor: an image augmentation library for machine learning,» 2017.
- [39] M. J. Garbade, «What is Google Colab?,» 15 enero| 2021. [En línea]. Available: <https://blog.education-ecosystem.com/what-is-google-colab/>. [Último acceso: 1 octubre| 2021].

- [40] S. Raschka y V. Mirjalili, *Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-Learn, and TensorFlow2*, Shangai, China: Copyright 2019 Packt Publishing Ltd., Birmingham, UK, 2019.
- [41] J. Giménez Campos, «Aplicaciones de GPU's en Visión e Inteligencia Artificial,» Universidad Politécnica de Valencia, Valencia, 2021.
- [42] T. Point, «Scikit Learn Introduction,» 2021. [En línea]. Available: https://www.tutorialspoint.com/scikit_learn/scikit_learn_introduction.htm. [Último acceso: septiembre 2021].
- [43] kapernikov, «image classification with scikit-learn,» 10 abril 2018. [En línea]. Available: <https://kapernikov.com/tutorial-image-classification-with-scikit-learn/>.
- [44] Keras, «Transfer learning y fine-tuning,» [En línea]. Available: https://keras.io/guides/transfer_learning/. [Último acceso: 06 09 2021].
- [45] K. Simonyan y A. Zisserman, «Very Deep Convolutional Networks for Large-Scale Image Recognition,» *Cornell University arXiv:1409.1556*, 2014.
- [46] E. C. Too, L. Yujian, S. Njuki y L. Yingchun, «A comparative study of fine-tuning deep learning models for plant disease identification,» *Computers and Electronics in Agriculture*, vol. 161, pp. 272-279, 2019.
- [47] K. Jarrett, K. Kavukcuoglu, M. Ranzato y Y. LeCun, «What is the best Multi-Stage Architecture for Object Recognition,» *IEEE 12th international conference on computer vision*, pp. 2146-2153, 2009.
- [48] S. Ioffe y C. Szegedy, «Batch normalization: Accelerating deep network training by reducing internal covariate shift.,» *International conference on machine learning*, pp. 448-456, 2015.
- [49] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens y Z. Wojna, «Rethinking the inception architecture for computer vision,» *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2828-2826, 2016.
- [50] C. Szegedy, «Inception-v4, inception-resnet and the impact of residual connections on learning,» de *Thirty-first AAAI conference on artificial intelligence.*, 2017.
- [51] «Network in network,» *arXiv:1312.4400*, 2013.
- [52] A. Aquino, J. M. Ponce, M. Borja, D. Tejada-Guzmán y J. M. Andújar, «Identificación y conteo de aceitunas en imágenes digitales tomadas en el olivar mediante morfología

- matemática y redes neuronales convolucionales,» de *XL Jornadas de Automática*, Coruña, Universidade da Coruña, Servizo de Publicacións, 2019, pp. 818-827.
- [53] G. Huang, Z. Liu y L. Van Der Maaten, «Densely connected convolutional networks,» *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700-4708, 2017.
- [54] K. He, X. Zhang, S. Ren y J. Sun, «Deep residual learning for image recognition,» *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016.
- [55] X. Zhang, J. Zou, K. He y J. Sun, «Accelerating very deep convolutional networks for classification and detection,» *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, nº 10, pp. 1943-1955, 2015.
- [56] S. V. Lab, «Imagenet,» 2020. [En línea]. Available: <https://www.image-net.org/>. [Último acceso: 06 09 2021].
- [57] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto y H. Adam, «Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.,» 2017.
- [58] D. Kingma y J. Ba, «Adam: A method for stochastic optimization,» *arXiv preprint arXiv:1412.6980*, 2014.
- [59] S. Ruder, «RUDER, Sebastian. An overview of gradient descent optimization algorithms,» *arXiv preprint arXiv:1609.04747*, 2016.
- [60] N. Qian, «On the momentum term in gradient descent learning algorithms,» *Neural networks*, vol. 12, nº 1, pp. 145-151, 1999.
- [61] T. Tieleman y G. Hinton, «Lecture 6.5-rmsprop: Divide the Gradient by a Running Average of Its Recent Magnitude,» *COURSERA: Neural networks for machine learning*, vol. 4, nº 2, pp. 26-31, 2012.
- [62] J. A. Gallardo Arancibia, «Metodología para el Desarrollo de Proyectos en Minería de Datos- CRISP-DM».
- [63] M. Grandini, E. Bagli y G. Visani, «Metrics for Multi-Class Classification: An Overview,» *arXiv preprint arXiv:2008.05756*, 2020.

- [64] K. H. Brodersen, C. S. Ong y K. Stephan, «The balanced accuracy and its posterior distribution,» *En 2010 20th international conference on pattern recognition*, pp. 3121-3124, 2010.
- [65] J. Opitz y S. Burst, «Macro F1 and Macro F1,» *arXiv:1911.03347*, 2019.
- [66] T. Davenport, J. Harris y R. Morison, *Analytics at Work: Smarter Decisions, Better Results*, Massachusetts: Harvard Business Press, 2009.

ANEXOS

Anexo I–Script compilación del modelo

```
x = Dense(5, activation='softmax', name='predictions')(x)
model_out = Model(base_model.input, x)
#We compile the model
model_out.compile(loss=keras.losses.categorical_crossentropy, optimizer=tf.keras.optimizers.Adam(learning_rate=INIT_LR, decay=INIT_LR / 100),metrics=['accuracy'])
model_out.summary()

callback = tf.keras.callbacks.EarlyStopping(monitor='loss', min_delta=0.0001, patience=3)

grado_train_dropout = model_out.fit(train_x, train_label, batch_size=batch_size,epochs=epochs,verbose=1,validation_data=(valid_X, valid_label),callbacks=[callback])
```

Anexo II – Script Codificación One-hot

```
#from keras.utils import to_categorical
from tensorflow.keras.utils import to_categorical
#Mezclar todo y crear los grupos de entrenamiento y testing
train_x,x_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=13)
print('Training data shape : ', train_x.shape, y_train.shape)
print('Testing data shape : ', x_test.shape, y_test.shape)
|
train_x = train_x.astype('float32')
x_test = x_test.astype('float32')
train_x = train_x / 255.
x_test = x_test / 255.

# Cambiar las etiquetas de la codificación categórica a la codificación one-hot
train_Y_one_hot = to_categorical(y_train)
test_Y_one_hot = to_categorical(y_test)

# Mostrar el cambio para la etiqueta de la categoría utilizando la codificación one-hot
print('Original label:', y_train[0])
print('After conversion to one-hot:', train_Y_one_hot[0])

train_x,valid_X,train_label,valid_label = train_test_split(train_x, train_Y_one_hot, test_size=0.2, random_state=13)

print(train_x.shape,valid_X.shape,train_label.shape,valid_label.shape)
```