



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO



Facultad de Ingeniería Civil, Sistemas y Arquitectura

Escuela Profesional de Ingeniería de Sistemas

TESIS

Aplicación Móvil usando redes neuronales convolucionales para la detección de enfermedades en las plantas de café en el distrito de Chirinos, provincia de San Ignacio, departamento de Cajamarca.

Para obtener el Título Profesional de:

Ingeniero de Sistemas

Flores Mendoza, Juan Carlos

Mejía Carhuajulca, Gianfranco

Autores

Mg. Ing. Arteaga Lora, Roberto Carlos

Asesor

LAMBAYEQUE-PERÚ

2023



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO



Facultad de Ingeniería Civil, Sistemas y Arquitectura
Escuela Profesional de Ingeniería de Sistemas

TESIS

Aplicación Móvil usando redes neuronales convolucionales para la detección de enfermedades en las plantas de café en el distrito de Chirinos, Provincia de San Ignacio, departamento de Cajamarca.

Para obtener el Título Profesional de:
Ingeniero de Sistemas

Aprobado por los miembros de Jurado:

Dr. Ing. Haro Maldonado, Edwar Ronald
Presidente del Jurado

Mg. Ing. Puican Gutiérrez, Robert Edgar
Secretario

Dr. Ing. Villegas Cubas, Juan Elías
Vocal

LAMBAYEQUE-PERÚ

2023



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO



Facultad de Ingeniería Civil, Sistemas y Arquitectura
Escuela Profesional de Ingeniería de Sistemas

TESIS

Aplicación Móvil usando redes neuronales convolucionales para la detección de enfermedades en las plantas de café en el distrito de Chirinos, Provincia de San Ignacio, Departamento de Cajamarca.

Para obtener el Título Profesional de:

Ingeniero de Sistemas

Flores Mendoza, Juan Carlos
Autor

Mejía Carhuajulca, Gianfranco
Autor

Mg. Ing. Arteaga Lora, Roberto Carlos
Asesor

LAMBAYEQUE-PERÚ

2023

DEDICATORIA

A mi madre, padre, hermano y abuelos, por el apoyo incondicional en cada paso que he dado, por darme la fortaleza y motivación para continuar y hacer realidad mis objetivos trazados.

Juan Carlos Flores Mendoza

A mi familia y a mi abuela Zoila, que me impulsan a seguir adelante y por ser parte importante en mi vida y hacer que todo sea posible.

Gianfranco Mejía Carhuajulca

AGRADECIMIENTOS

A Dios, por ser mi guía y acompañante en mi vida. A nuestro asesor de tesis por brindarnos sus conocimientos y apoyarnos en la elaboración de esta investigación.

Juan Carlos Flores Mendoza

Agradezco a Dios por guiar mis pasos, a nuestro asesor de tesis, maestros por sus enseñanzas para desarrollarme profesionalmente y haberme brindado sus conocimientos.

Gianfranco Mejía Carhuajulca

RESUMEN

La presente investigación se desarrolló en el distrito de Chirinos, provincia de San Ignacio, departamento de Cajamarca, el problema que se encontró fue que los agricultores de la zona tienen la dificultad de detectar las enfermedades que afectan a sus plantaciones de café.

Se construyó un modelo de detección de enfermedades en plantas de café, basándose en redes neuronales convolucionales con procesamiento de imágenes utilizando un total de 34,637 imágenes de las 4 enfermedades más comunes y propias de la zona.

El modelo funcionó teniendo una tasa de acierto de 99.6%, con una función de pérdida de 0.016%, se comprobó con una muestra de 72 imágenes aleatorias existentes en 1 hectárea de café, y el diagnóstico dado por un experto agrónomo con esta misma muestra de plantas de café. Finalmente, la aplicación móvil y el modelo de red neuronal convolucional funcionaron correctamente.

Palabras claves: Red neuronal convolucional, enfermedad del café, procesamiento de imágenes, aplicación móvil.

ABSTRACT

The present research was developed in the district of Chirinos, province of San Ignacio, department of Cajamarca, the problem that was found was that farmers in the area have difficulty detecting diseases that affect their coffee plantations.

A model for detecting diseases in coffee plants was built based on convolutional neural networks with image processing using a total of 34,637 images of the 4 most common diseases in the area.

The model worked with an accuracy rate of 99.6%, with a loss function of 0.016%, tested with a sample of 72 random images existing in 1 hectare of coffee, and the diagnosis given by an agronomist expert with this same sample of coffee plants. Finally, the mobile application and the convolutional neural network model worked correctly.

Key words: Convolutional neural network, coffee disease, image processing, mobile application.

ÍNDICE

CAPÍTULO I: MARCO LÓGICO	14
1.1 Situación problemática.....	14
1.2 Formulación del problema	16
1.3 Hipótesis	16
1.4 Objetivos de la investigación	16
1.4.1 Objetivo general	16
1.4.2 Objetivo específico.....	16
CAPÍTULO II: MARCO TEÓRICO.....	17
2.1 Antecedentes	17
2.1.1 Antecedentes a nivel Internacional.....	17
2.1.2 Antecedentes a Nivel Nacional.....	20
2.1.3 Antecedentes a Nivel Local.....	21
2.2 Bases teóricas.....	22
2.2.1 El café.....	22
2.2.2 Paisaje Cafetalero Peruano	22
2.2.3 Métodos de información.....	23
2.2.4 Enfermedades del Café.....	24
2.2.5 Tipos de agroquímicos.....	25
2.2.6 Control biológico.....	27
2.2.7 Visión artificial	28
2.2.8 Procesamiento de imágenes digitales	28
2.2.9 OpenCv.....	30
2.2.10 Machine Learning.....	31
2.2.11 Visión general de los aprendizajes del aprendizaje automático	31
2.2.12 Redes Neuronales	32
2.2.13 Redes neuronales convolucionales	33
2.2.14 TensorFlow	37
2.2.15 Keras.....	37
2.2.16 Aplicaciones móviles.....	37
2.2.17 Lenguaje para desarrollo en Android	37
CAPÍTULO III: MARCO METODOLÓGICO	39
3.1 Tipo de investigación.....	39

3.2	Variables	39
3.2.1	Variable dependiente	39
3.2.2	Variable independiente	39
3.3	Tabla de operacionalización de variable	40
3.4	Población y muestra	41
3.4.1	Población	41
3.4.2	Muestra	41
3.5	Técnicas, instrumentos de recolección de datos	42
3.6	Diseño de contrastación de hipótesis	43
3.7	Metodología	44
CAPÍTULO IV: DESARROLLO.....		46
4.1	Generalidades del distrito de Chirinos	46
4.1.1	Ubicación Geográfica	46
4.2	Requisitos y preparación	47
4.2.1	Requisitos previos	47
4.2.2	Preparación del entorno de trabajo	47
4.3	Procesamiento de datos	48
4.4	Descripción de los conjuntos de imágenes	49
4.5	Funcionamiento de la red	49
4.5.1	Metodología de la red neuronal convolucional	50
4.5.2	Data augmentation	51
4.6	Modelo	52
4.6.1	Datos	52
4.6.2	Arquitectura	53
4.6.3	Entrenamiento	58
4.6.4	Evaluación	61
4.6.5	Predicción	62
4.7	Android Studio	63
4.7.1	Barra de Navegación	63
4.7.2	Firestore Storage	64
4.7.3	Firestore Database	65
4.7.4	Procesamiento del Modelo Entrenado	66
4.8	Firestore	67

4.8.1	Proyecto Firebase	67
4.8.2	Firestore Storage	68
4.8.3	Firestore Database	69
4.8.4	WebServices	70
4.9	Aplicación Móvil	71
4.9.1	Interfaz Gráfica.....	71
4.9.2	Formulario y análisis de datos	72
CAPÍTULO V: RESULTADOS.....		75
CAPÍTULO VI: DISCUSIÓN		81
CAPÍTULO VII: CONCLUSIONES		83
CAPÍTULO VIII: RECOMENDACIONES		84
REFERENCIAS BIBLIOGRÁFICAS		85
ANEXOS		87

ÍNDICE DE TABLAS

Tabla 1	Antecedente Internacional 01	17
Tabla 2	Antecedente Internacional 02	18
Tabla 3	Antecedente Internacional 03	19
Tabla 4	Antecedente Nacional 01	20
Tabla 5	Antecedente Local 01	21
Tabla 6	Enfermedades del Café.....	24
Tabla 7	Agroquímicos para la Enfermedad Mancha Hierro.....	25
Tabla 8	Agroquímicos para la Enfermedad Roya de Hoja.....	26
Tabla 9	Agroquímicos para la Enfermedad Minador de Hoja.....	26
Tabla 10	Agroquímicos para la Enfermedad Phoma de Hoja	27
Tabla 11	Tabla de operacionalización de Variable.....	40
Tabla 12	Técnicas, Instrumentos de Recolección de Datos	42
Tabla 13	Contrastación de Hipótesis	43
Tabla 14	Comparativa de Resultados entre el Experto Agrónomo y la Aplicación Móvil	75
Tabla 15	Valores de los Hiperparámetros	75
Tabla 16	Variación de la Precisión y del Valor de Pérdida con la Época	77
Tabla 17	Comparativa de Resultados entre el Experto Agrónomo y la Aplicación Móvil	78

ÍNDICE DE FIGURAS

Figura 1	Enfermedades del Café	25
Figura 2	Procesamiento de Imágenes Digitales	29
Figura 3	Aprendizaje Automático	32
Figura 4	Redes Neuronales	33
Figura 5	Redes Neuronales Convolucionales	34
Figura 6	Capa de Convolución	35
Figura 7	Max Pooling y Average Pooling	36
Figura 8	Capa Flatten	36
Figura 9	Metodología de la Investigación	44
Figura 10	Metodología de la Investigación	45
Figura 11	Ubicación Geográfica del Distrito de Chirinos	46
Figura 12	Directorios de Trabajo	47
Figura 13	Dataset de las Hojas del Café.....	48
Figura 14	Metodología de la Red Neuronal Convolutiva	50
Figura 15	Data Augmentation	51
Figura 16	Clasificación de las Enfermedades de la Hoja del café por categoría.....	52
Figura 17	Función Relu.....	53
Figura 18	Arquitectura por capas	54
Figura 19	Capas del Modelo	55
Figura 20	Modelo Secuencial	56
Figura 21	Algoritmos de Optimización	57
Figura 22	Métrica de Precisión	58
Figura 23	Early Stopping	59
Figura 24	Entrenamiento con 20 Épocas	59
Figura 25	Precisión entre los Datos de Entrenamiento y Validación	60
Figura 26	Pérdida entre los Datos de Entrenamiento y Validación	60
Figura 27	Evaluación con los Datos de Prueba	61
Figura 28	Matriz de Confusión	62
Figura 29	Codificación de la Barra de Navegación	63
Figura 30	Codificación de Firebase Storage.....	64
Figura 31	Codificación de Firebase Database	65
Figura 32	Vinculación del Api Modelo con Android	66
Figura 33	Creación del Proyecto en Firebase	67
Figura 34	Procesamiento de Datos en Firebase Storage	68
Figura 35	Procesamiento de Datos en Firebase Database.....	69
Figura 36	WebServices	70
Figura 37	Interfaz Gráfica de la Aplicación Scan Plant.....	71
Figura 38	Formulario de la Aplicación Scan Plant.....	72
Figura 39	Análisis de Datos de la Aplicación Scan Plant.....	73
Figura 40	Wiki de la Aplicación Scan Plant.....	74
Figura 41	Resultado del análisis del experto agrónomo	76
Figura 42	Resultado del análisis de la Aplicación Móvil	76
Figura 43	Respuesta 1 del Anexo-03.....	93
Figura 44	Respuesta 2 del Anexo-03.....	93

Figura 45	Respuesta 3 del Anexo-03.....	94
Figura 46	Respuesta 4 del Anexo-03.....	94
Figura 47	Respuesta 5 del Anexo-03.....	95
Figura 48	Respuesta 6 del Anexo-03.....	95
Figura 49	Respuesta 7 del Anexo-03.....	96
Figura 50	Respuesta 8 del Anexo-03.....	96
Figura 51	Respuesta 9 del Anexo-03.....	97
Figura 52	Respuesta 10 del Anexo-03.....	97

CAPÍTULO I: MARCO LÓGICO

1.1 Situación problemática

A nivel mundial, el café es una de los principales productos agrícolas más comercializados en el mundo, ocupando el segundo lugar por detrás del petróleo, en el mercado mundial del café es muy dinámico, su oferta y demanda crecen en los mercados, aproximadamente 50 países producen café, Brasil, vietnam y Colombia lideran la producción, siendo la unión europea, estados unidos y Japón sus principales mercados. (Proyecto Cafe y Clima, 2017, pág. 17).

En América latina, Según (Minagri, 2019) afirma que “la baja producción del café es causa de la enfermedad de la roya y el cambio climático vienen afectando a la comunidad cafetalera en donde se proyecta para la campaña 2021-2022 bajas cosechas del café”.

El Perú, “Es uno de los principales exportadores y productores de café del agro tradicional” (Zegarra M, 2019). “Actualmente el Perú se ubica entre los principales exportadores de café a nivel mundial” (JNC, Junta del Café, 2021). El cultivo de café en el Perú representa un aproximado de 223 mil familias que se dedican a la siembra de café, sin embargo, hay siete regiones que concentran con un total de 91% de productores que entre ellas tenemos las regiones de Huánuco, Pasco, Cusco, Amazonas, Junín, Cajamarca, San Martín. (Díaz Vargas & Carmen Willems, 2017, pág. 19).

Actualmente, el eje de la comercialización y eje norte de la producción de café son las provincias de san Ignacio y jaén en Cajamarca, las cuales se han consolidado como

los ejes más importantes del sector de producción de café representando un total del 46.5% del valor FOB. (JNC, Exportación de café enero - diciembre, 2021).

En el distrito de Chirinos, provincia de San Ignacio, Departamento de Cajamarca, el dueño de la empresa cafetalera EXPORTADORA DE CAFÉ CASMAR E.I.R.L, y el ingeniero agrónomo encargado de la producción de café, nos comentaron sobre las deficiencias que presentan algunos agricultores de la zona para detectar las enfermedades de café mediante el uso de la percepción visual en la cual tienden a confundir las enfermedades y trae como consecuencias gastos excesivos en la producción de café y el uso erróneo de los agroquímicos causando la muerte de la planta de café.

La presente investigación tiene como importancia, la implementación de un sistema de inteligencia artificial que usará redes neuronales convolucionales, para apoyar a los agricultores en el tratamiento de la planta de café, y permitirá tomar mejores decisiones a los agricultores en la detección de enfermedades teniendo una mayor información de cada una de ellas para un mejor uso y correcto de agroquímicos.

Puesto que este trabajo de investigación tiene una relevancia social, donde muchos agricultores de café, necesitan un apoyo para tratar las enfermedades en sus plantaciones de café y tomar mejores decisiones para la mejora de su producción y cosecha de café.

Finalmente, el sistema de inteligencia artificial va a permitir a los agricultores de café, mejorar en el tratamiento de las enfermedades en sus plantaciones de café, teniendo una información detallada para el uso correcto de agroquímicos.

1.2 Formulación del problema

¿De qué manera un aplicativo móvil que usa redes neuronales convolucionales permitirá apoyar a los agricultores en el tratamiento de las enfermedades en sus plantas de café?

1.3 Hipótesis

Implementar redes convolucionales y un sistema de inteligencia artificial, permitirá apoyar en la detección de las enfermedades en las plantas de café en el distrito de Chirinos.

1.4 Objetivos de la investigación

1.4.1 Objetivo general

- Mejorar en la toma de decisiones en el tratamiento de enfermedades que afectan a las plantaciones de café en el distrito de Chirinos con una aplicación móvil que usa redes neuronales convolucionales.

1.4.2 Objetivo específico

- Describir adecuadamente los tipos de enfermedades que afectan a las plantas de café en el distrito de Chirinos.
- Optimizar los tiempos en la identificación y búsqueda de enfermedades en las plantaciones de café.
- Determinar el tratamiento adecuado utilizando agroquímicos en las enfermedades identificadas en las plantas de café.

CAPÍTULO II: MARCO TEÓRICO

2.1 Antecedentes

2.1.1 Antecedentes a nivel Internacional

Tabla 1

Antecedente Internacional 01

TÍTULO	<i>“Análisis de imágenes multiespectrales para la detección de cultivos y detección de plagas y enfermedades en la producción de café”</i>
PAÍS	México
UNIVERSIDAD	Instituto Tecnológico Superior de Misantla
FECHA	2018
AUTOR(ES)	Arely Guadalupe Sánchez-Méndez, Simón Pedro Arguijo-Hernández
OBJETIVO	Desarrollar un método que permita detectar las enfermedades y plagas en la producción de los cultivos de café, usando imágenes multiespectrales del satélite Landsat8.
SÍNTESIS DE LA RELACIÓN CON LA PRESENTE INVESTIGACIÓN	<p>El desarrollo de su investigación tiene una relación con nuestro caso de estudio por la utilización de los siguientes procesos: pre procesamiento de imágenes, la interpretación digital que es analizada mediante un proceso de clasificación supervisada.</p> <p>En resumen, los resultados que se realizaron del muestreo en los campos de entrenamiento, tuvieron un valor aceptable, en donde el valor del índice de vegetación de diferencia normalizada (NDVI), se encuentra a un 0,2 a 0,8 mientras que el índice normal debe estar entre 0,4 a 0,9 dependiendo del área de cultivo y del índice de área foliar, en donde se validó y se evaluaron los resultados mediante una matriz de error.</p>

Fuente: (Sánchez Méndez & Arguijo Hernández, 2018)

Tabla 2*Antecedente Internacional 02*

TÍTULO	<i>“Aplicación móvil de detección y clasificación de “la roya” en hojas de café robusta mediante aprendizaje automático”</i>
PAÍS	Ecuador
UNIVERSIDAD	Escuela superior politécnica agropecuaria de Manabí Manuel Félix López
FECHA	2019
AUTOR(ES)	Cusme Zambrano, Kevin Daniel; Loor Pinargote, Angélica María
OBJETIVO	Desarrollar una aplicación móvil para automatizar la detección y automatización de la “Roya” en las hojas de café, usando el procesamiento de imágenes para determinar cuándo una hoja estaba infectada de roya.
SÍNTESIS DE LA RELACIÓN CON LA PRESENTE INVESTIGACIÓN	<p>Se toma en consideración este trabajo de investigación, porque aplica técnicas de reconocimiento de imágenes, que se usa para identificar la plaga de la roya en los campos de cultivo del café.</p> <p>En conclusión, se desarrolló una aplicación y se implementó un algoritmo para detectar la plaga de la roya en los cultivos de café, dando un 95% de efectividad en la roya.</p>

Fuente: (Cusme Zambrano & Loor Pinargote, 2019)

Tabla 3*Antecedente Internacional 03*

TÍTULO	<i>“Disease Detection in Coffee Plants Using Convolutional Neural Network”</i>
PAÍS	India
UNIVERSIDAD	Delhi Technological University
FECHA	2020
AUTOR(ES)	Manoj Kumar, Pranav Gupta, Puneet Madhav, Sachin
OBJETIVO	Contribuir con los agricultores en la lucha contra las enfermedades de café, que resulta en el aumento de la producción de café en la India. Las redes neuronales convolucionales (CNN) han demostrado su eficacia y precisión en la clasificación de imágenes y reconocimiento de patrones.
SÍNTESIS DE LA RELACIÓN CON LA PRESENTE INVESTIGACIÓN	<p>Esta investigación es muy importante, porque se va a utilizar la misma metodología para tener una mejora de eficacia y precisión en el reconocimiento de imágenes.</p> <p>Asimismo, se propone un modelo de red neuronal de convolución, de aprendizaje de transferencia, reduciendo el tiempo de entrenamiento del modelo de forma significativa.</p> <p>En conclusión, los resultados logrados tienen una tasa de éxito, en donde se aplicó la técnica de aumento de datos para ampliar el conjunto de datos utilizado para entrenar la red, este modelo propuesto ha logrado una alta precisión del 97.61%.</p>

Fuente: (Kumar, Gupta, Madhav, & Sachin, 2020)

2.1.2 Antecedentes a Nivel Nacional

Tabla 4

Antecedente Nacional 01

TÍTULO	<i>“Aplicación de imágenes hiperespectrales para la detección temprana de roya amarilla (hemileya vastatrix) en café (coffea arábica), en el distrito de Limbamba, provincia Rodríguez de Mendoza región Amazonas”</i>
PAÍS	Distrito de Limbamba, provincia Rodríguez de Mendoza región Amazonas
UNIVERSIDAD	Universidad Nacional Toribio Rodríguez de Mendoza
FECHA	2019
AUTOR(ES)	López Portocarrero, César Augusto
OBJETIVO	Desarrollar una metodología para el análisis que causa la roya amarilla, en la cual se usó la técnica de imágenes hiper espectrales para determinar la severidad del daño de la roya.
SÍNTESIS DE LA RELACIÓN CON LA PRESENTE INVESTIGACIÓN	<p>Se toma en consideración esta investigación por que usa la clasificación de árbol de decisiones, donde se ofrece una mayor precisión en sus análisis objetivos cuantitativos para determinar la fiabilidad de las infecciones tempranas en caso de la roya amarilla en las plantaciones de café.</p> <p>Por consiguiente, se utiliza un método de clasificación aplicando una metodología de árbol de decisión en la que se compara con la técnica de hiper espectrales en el software matemático de Matlab V2010.</p> <p>En conclusión, se logró determinar que existe incongruencia de fiabilidad entre ambos métodos, en donde las muestras evaluadas con imágenes hiper espectrales se muestra como una técnica poco confiable para detectar infecciones en el tejido de las hojas causada por la enfermedad de la roya amarilla, siendo 0,5% mejor la técnica de la clasificación por árbol de decisiones.</p>

Fuente: (López Portocarrero, 2019)

2.1.3 Antecedentes a Nivel Local

Tabla 5

Antecedente Local 01

TÍTULO	<i>“Sistema de visión artificial para apoyar en la identificación de plagas y enfermedades del cultivo de sandía en el distrito de Ferreñafe”</i>
PAÍS	Perú
UNIVERSIDAD	Universidad Santo Toribio de Mogrovejo
FECHA	2019
AUTOR(ES)	Piscoya Ferreñan, Jesus Enrique
OBJETIVO	Identificar que tipos de plagas y enfermedades infectan a la sandía, para ayudar a los agricultores a realizar la identificación de las plagas y enfermedades, en la cual demuestra que con su aplicación le muestra los resultados en 4 segundos aproximadamente.
SÍNTESIS DE LA RELACIÓN CON LA PRESENTE INVESTIGACIÓN	<p>Esta investigación es un antecedente directo por el uso de la inteligencia artificial y por la implementación de un sistema para la ayuda a los agricultores y aplica un procesamiento de imágenes para la identificación de las plagas y las enfermedades de la sandía.</p> <p>En Conclusión, para este trabajo de investigación se demuestra que la visión artificial ayudó a los agricultores un 82.6% a tener una mejor información de las enfermedades y plagas, disminuyendo los costos para el proceso de identificación un 43.48%, concluyendo que el sistema es rentable.</p>

Fuente: (Piscoya Ferreñan, 2019)

2.2 Bases teóricas

2.2.1 *El café*

El café (*coffea*), es un grano que se cosecha de un arbusto llamado cafeto, crece en climas cálidos, siendo una de las bebidas por su consumo más populares a nivel mundial, sin embargo, los cafetos son muy vulnerables a diversas plagas y enfermedades. (Balardin, 2010).

El uso inexperto y el descuido de los agricultores en el uso de los plaguicidas pueden dar lugar al desarrollo de enfermedades en sus cultivos, afectando gravemente el desarrollo y crecimiento del cafeto, por ello, la detección de las enfermedades de las plantas es un aspecto muy esencial para la agricultura (Miller, Beed, & Lapaire Harmon, 2009, pág. 15).

2.2.2 *Paisaje Cafetalero Peruano*

El café en el Perú es un cultivo que necesita un clima cálido y se siembra en los valles interandinos y se extiende por la cordillera de los andes, siendo el café arábica y café robustas, los que mayormente se comercializan a nivel global, siendo el café arábico el más cosechado a nivel mundial, representado con un 60% (Di Donfrancesco, Gutierrez Guzman, & Chambers IV, 2019).

En el Perú el café uno de los productos de mayor exportación, el café se presenta en 16 regiones del Perú, en la cual solo 5 regiones concentran un 86% en la producción de café, siendo estas regiones Cajamarca, Junín, Amazonas, San Martín y Cusco. (Minagri, 2019).

Según (CPCC, 2021), El Perú exporta principalmente 3 tipos de café, entre ellos tenemos el café tradicional, el café certificado y el café premium (café de alta pureza) siendo promovido por el sector cafetalero en los cuales los productos peruanos obtuvieron galardones.

El principal eje comercial de café del norte del Perú se inicia en el departamento de Cajamarca, siendo las provincias de san Ignacio y jaén, en donde se han reconocido un total de 36 organizaciones consolidándose en el 2019 como las 2 provincias que representan un 46.5% de mayor valor FOB. (JNC, 2020).

2.2.3 Métodos de información

Hoy en día, la computación blanda desempeña un papel muy importante en la detección y precisión de enfermedades que atacan a una hoja de una planta de café, se utilizan diversas técnicas como algoritmos genéticos, aprendizaje automático, el uso de la visión artificial y el procesamiento de imágenes. (Abu Mettleq & Abu-Naser, 2019).

Según (Derwin, Wahyu, Miranty, & Muhammad, 2013), “el procesamiento de imágenes y el uso de las redes neuronales artificiales pueden lograr una tasa de éxito hasta de un 90%”.

Deep Learning, ha dominado el campo de la clasificación de imágenes durante los últimos años, por lo que la red neuronal de convolución propuesta se usará para detectar las enfermedades de las hojas de café y categorizarlas en 5 clases: sanas, hojas enfermas con manchas de Cercospora, Phoma, roya de la hoja del café (CLR) y minador de la hoja con una alta tasa de éxito.

2.2.4 Enfermedades del Café

Tabla 6

Enfermedades del Café

Nombre Común	Agente Causante	Síntomas
Minador de Hoja (Leucoptera Coffeella)	Polilla Perileucoptera coffeella	<ul style="list-style-type: none"> • Minaduras en forma de ampolla con verdes, pero se vuelven color marrón oscuro. • Causa desopilación y pérdida del grano
Phoma de Hoja (Phoma Costarricensis)	Phoma costarricensis	<ul style="list-style-type: none"> • Manchas de color negro que se rodea el borde la hoja. • Muerte de las ramificaciones y de las hojas.
Roya de hoja (Hemileia vastratrix Berkeley & Broome)	Hongo Hemileia vastatrix	<ul style="list-style-type: none"> • Manchas al tocarlas desprenden un polvo de color naranja. • Defoliación y muerte de la hoja ante fuertes infestaciones.
Mancha de Hierro (Cercospora Coffeicola)	Mycosphaerella coffeicola	<ul style="list-style-type: none"> • Las hojas muestran manchas circulares con un centro de color marrón. • Las hojas infectadas se vuelven amarillas provocando una defoliación prematura.
Hoja Saludable	-	<ul style="list-style-type: none"> • Hoja Saludable

Fuente: Elaboración propia

Figura 1

Enfermedades del Café



Fuente: Elaboración propia

2.2.5 Tipos de agroquímicos

2.2.5.1 Mancha de hierro

En la presencia de esta enfermedad se deberá usarse los siguientes productos:

Tabla 7

Agroquímicos para la Enfermedad Mancha Hierro

Producto	Dosis
Benlate (50%)	1g/litro de agua
Dithane M-45 (80%)	3g/litro de agua
Oxicloruro de cobre (50%)	5g/litro de agua

Fuente: (Almengor, 2019)

2.2.5.2 Roya de hoja

En la presencia de esta enfermedad, es posible tener un control eficiente usando los siguientes funguicidas:

Tabla 8

Agroquímicos para la Enfermedad Roya de Hoja

Producto	Dosis
Oxicloruro de cobre (50%) a 3.5 kg/ha	6 g/litro de agua
Ocidos de cobre (50%) o hidrocidos de cobre (50%) a 2.5 kg/ha	g/litro de agua

Fuente: (Almengor, 2019)

2.2.5.3 Minador de hoja

Para la presencia de esta enfermedad, se pueden utilizarse los siguientes insecticidas.

Tabla 9

Agroquímicos para la Enfermedad Minador de Hoja

Producto	Dosis
Clorpirifos	0.5 litros más 0.5 litros de aceite mineral en 200 litros de agua
Decis 10Ec	100 mililitros más 0.5 litros de aceite mineral en 200 litros de agua

Fuente: (Almengor, 2019)

2.2.5.4 Phoma de Hoja

Para la presencia de esta enfermedad se pueden utilizar los siguientes productos.

Tabla 10

Agroquímicos para la Enfermedad Phoma de Hoja

Producto	Dosis
Captan	1g/litro de agua
Benlate (50%)	1g/litro de agua

Fuente: (Almengor, 2019)

2.2.6 Control biológico

2.2.6.1 Mancha de hierro

El problema de esta enfermedad se puede prevenir mediante las siguientes observaciones:

- La fertilización de la planta adecuada
- Un control adecuado de nematodos fitoparásitos
- Evitar el exceso de iluminación

2.2.6.2 Roya del café

Para tener un mayor control de esta enfermedad se puede prevenir siguiendo las siguientes instrucciones:

- Usar fertilizantes adecuados y verificar un mejor control del suelo en el pH.
- Sembrar en lugares de sombras para así evitar un estrés de la planta y proteger de la humedad del suelo.

2.2.6.3 Minador de hoja

El problema de esta enfermedad se puede solucionar con las lluvias ya que es un medio natural y el agua inunda las hojas y las larvas de la hoja mueren ahogadas.

2.2.6.4 Phoma de la hoja

El problema de esta enfermedad se puede prevenir siguiendo el manejo correcto del cultivo:

- El uso de sombras temporal para el cafeto.
- Nutrición adecuada de abonos.
- Detección temprana de enfermedad en la hoja.

2.2.7 Visión artificial

Según (Amao Sullcahuaman, y otros) afirma que la visión artificial es la utilización de las técnicas adecuadas de las cuales permiten el análisis y el procesamiento de cualquier tipo de información en el campo de la inteligencia artificial a través de imágenes digitales. Los procesos que utiliza la visión artificial para realizar los análisis de las imágenes son los siguientes procesos: adquisición de la imagen, preprocesamiento, segmentación, representación y descripción, reconocimiento e interpretación.

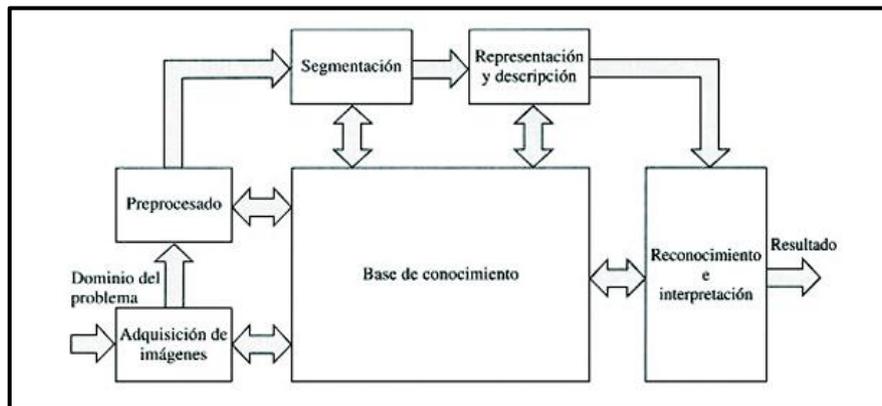
2.2.8 Procesamiento de imágenes digitales

Según (Parra Plaza, 2015) define, “el procesamiento de imágenes digitales como la representación de diferentes estrategias y métodos que se utilizan para analizar, modificar y/o extraer cualquier tipo de información”.

Para (Mejía Vilet, 2005) afirma que “el procesamiento de imágenes digitales representa un papel muy importante porque está construido sobre bases probabilísticas y matemáticas ya que está basado para desarrollar juicios visuales subjetivos para tener un mejor entendimiento con la percepción humana”.

Figura 2

Procesamiento de Imágenes Digitales



Fuente: (Parra Plaza, 2015)

2.2.8.1 Adquisición de la imagen

En esta etapa se captura la muestra mediante algún dispositivo o algún sensor y en la cual la calidad de las imágenes tomadas depende de la precisión de la digitalización de este dispositivo (Parra Plaza, 2015).

2.2.8.2 Preprocesamiento

“En esta etapa posterior a la adquisición de la imagen se trata de mejorar la calidad de la imagen utilizando técnicas como la mejora en el contraste de imagen” (Parra Plaza, 2015).

2.2.8.3 Segmentación

“Diríamos que esta etapa es la más importante porque en este proceso se aplican los algoritmos que se va a utilizar para separar la imagen en sus componentes, buscando así una mejora en la búsqueda de resultados” (Parra Plaza, 2015).

2.2.8.4 Representación y descripción

“En esta etapa se permite obtener la información de interés y adecuada para la identificación de patrones y rasgos, por ejemplo, las manchas de las hojas” (Parra Plaza, 2015).

2.2.8.5 Reconocimiento e interpretación

“Finalmente, en la última etapa se basa principalmente en la interpretación de resultados de las fases anteriores comparándolos con referentes en base de datos, buscando similitudes en correspondencias basadas en patrones” (Parra Plaza, 2015).

2.2.9 *OpenCv*

OpenCv (Open Computer Vision), “es una librería que se usa para el procesamiento de imágenes, que son usadas principalmente para aplicaciones de visión por el computador en tiempo real, de esta manera ayuda a detectar y reconocer objetos” (Mogena Soler, 2014)

Esta librería cuenta con un amplio conjunto de algoritmos de visión por computador y aprendizaje automático con un más de 2500 algoritmos optimizados, entre las principales funcionalidades de opencv tenemos las siguientes:

- Procesamiento de imágenes (Brillo, Contraste, Umbral)
- Detección de objetos
- Captura de imágenes en tiempo real.

Entre los principales lenguajes de programación es compatible con C++, C, Python, Java, y los sistemas operativos de Linux, Windows, Android y macOS.

2.2.10 Machine Learning

Machine learning o aprendizaje automático, es un proceso computacional que para lograr una tarea deseada se debe utilizar datos de entrada sin estar codificado, logrando un resultado concreto. Estos algoritmos se vuelven mejores mediante la repetición para lograr una tarea deseada adaptándose a su arquitectura. (El Naqa & J.Murphy, 2015).

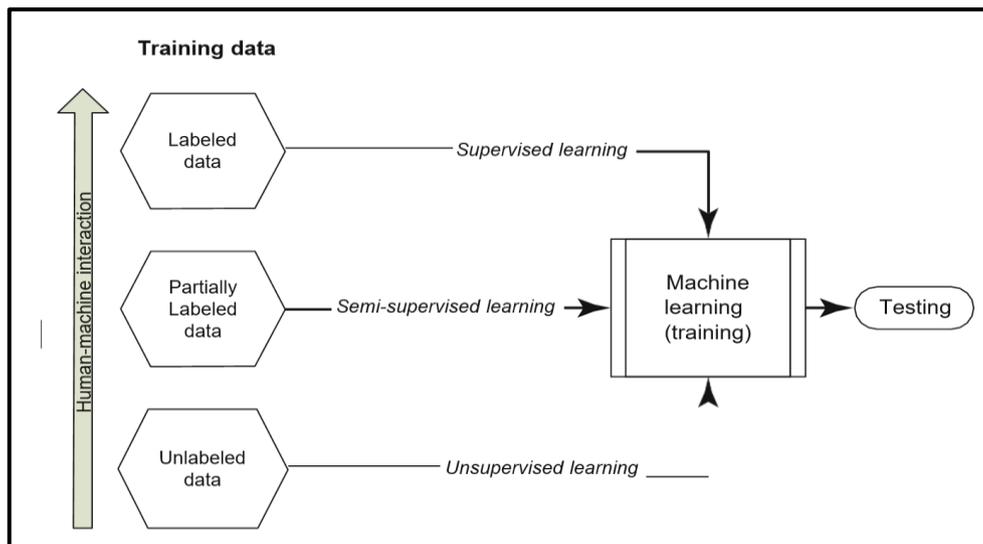
El objetivo de este algoritmo es optimizar los datos de entrada para producir un resultado deseado partiendo de datos no vistos, el entrenamiento es la parte fundamental del aprendizaje automático en la cual es un buen algoritmo que a medida que entrena aprende de sus errores mientras procesa los datos.

2.2.11 Visión general de los aprendizajes del aprendizaje automático

- **El aprendizaje automático:** puede dividirse por su naturaleza de etiquetado de las siguientes formas:
- **Aprendizaje supervisado:** Es usado para mapear datos de entrada a partir de muestras desconocidas donde los datos de salida son etiquetados.
- **Aprendizaje no supervisado:** Es usado para la entrada de datos al sistema de aprendizaje.
- **Aprendizaje semi-supervisado:** está formado por una parte de datos que parcialmente está etiquetada se utiliza para inferir datos no etiquetados, es la unión entre un aprendizaje supervisado y uno no supervisado.

Figura 3

Aprendizaje Automático



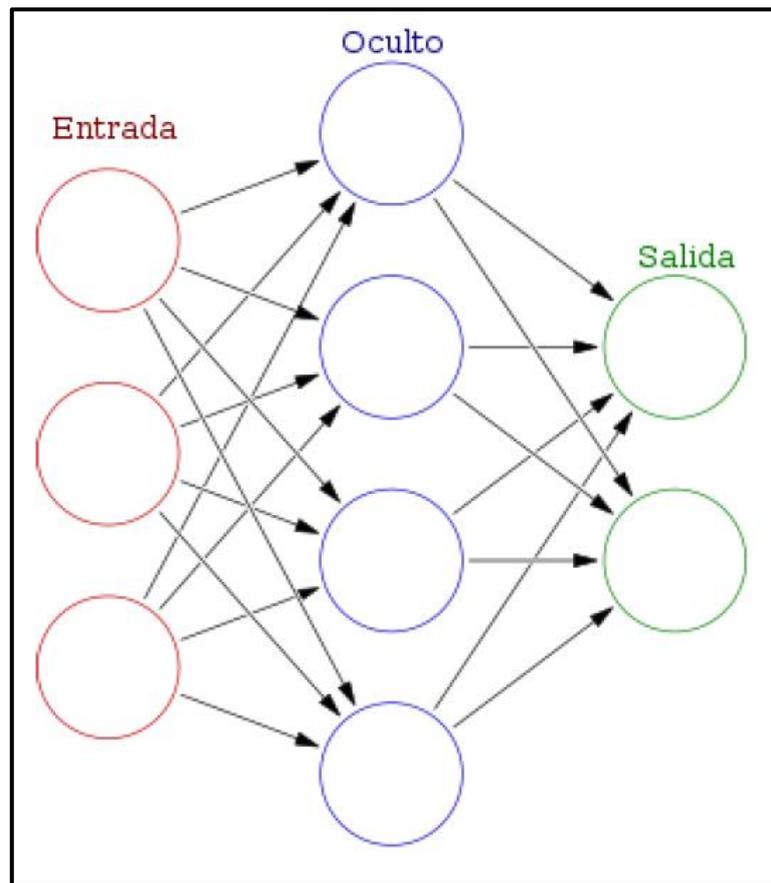
Fuente: (El Naqa & J.Murphy, 2015)

2.2.12 Redes Neuronales

Las redes neuronales es un modelo computacional que están inspirados en el cerebro humano en la cual van a recibir datos de entrada y cada uno de estos datos de entrada llegan a un nodo que se llama neurona, en la cual las neuronas son redes que son agrupadas en capas que conforman la red neuronal. Las neuronas de la red están etiquetadas mediante un valor numérico en lo cual se van a modificar los datos de entrada, los nuevos valores van a recorrer obtenidos en la red neuronal van a recorrer el camino de la red hasta alcanzar el final de la red donde se obtiene una predicción que será calculada por la red (Llano Carmona, 2021).

Figura 4

Redes Neuronales



Fuente: (Llano Carmona, 2021)

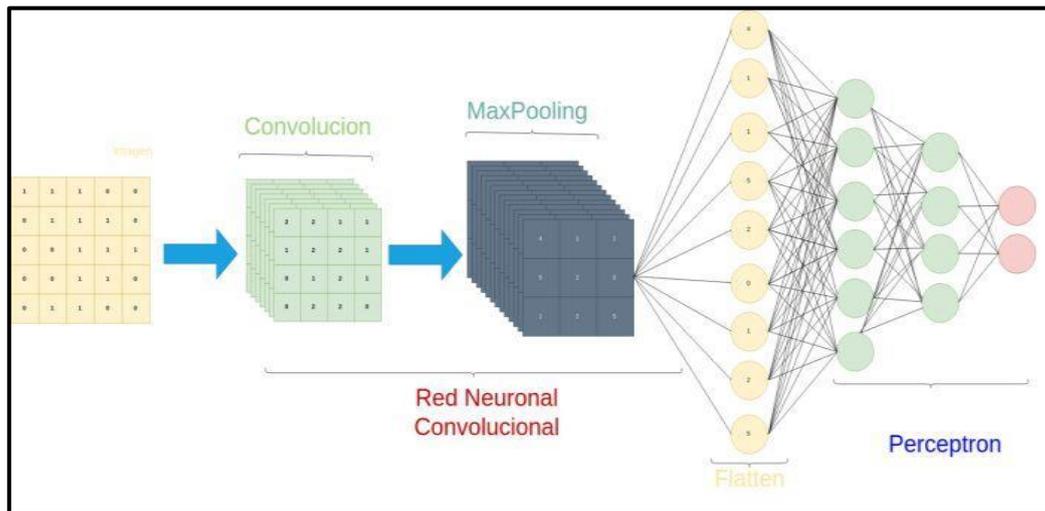
2.2.13 Redes neuronales convolucionales

Las redes neuronales convolucionales (CNN) son usadas para trabajar con imágenes ya que son un tipo de algoritmo de *Deep learning*, son capaces de detectar objetos simples en las primeras capas, y con los procesos de entrenamiento poco a poco se van especializando en reconocer formas u objetos más complejos. (Llano Carmona, 2021).

Para las extracciones de características de los objetos se utiliza la composición de capas convolucionales, capas de agrupación y capas totalmente conectadas.

Figura 5

Redes Neuronales Convolucionales

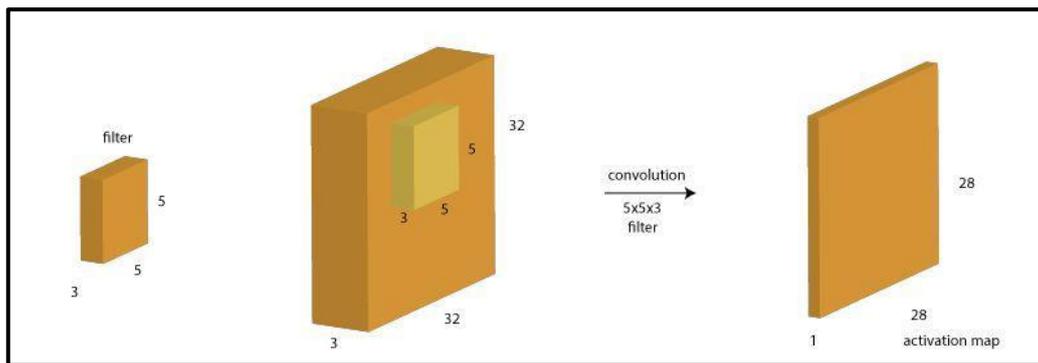


Fuente: (Llano Carmona, 2021)

Las capas de convolución se realizan por la multiplicación de la matriz de datos de entrada por una matriz bidimensional de filtros ya que son operaciones lineales, el filtro se utiliza para detectar características en la parte de la entrada de datos, en la cual dicho filtro permite el desplazamiento de derecha a izquierda y de arriba hacia abajo obteniendo un resultado de una matriz bidimensional o también llamada *feature map*, es muy importante el entrenamiento del modelo de nuestra red neuronal porque aprenderá cuales son los valores adecuados para nuestro filtro (Llano Carmona, 2021).

Figura 6

Capa de Convolución



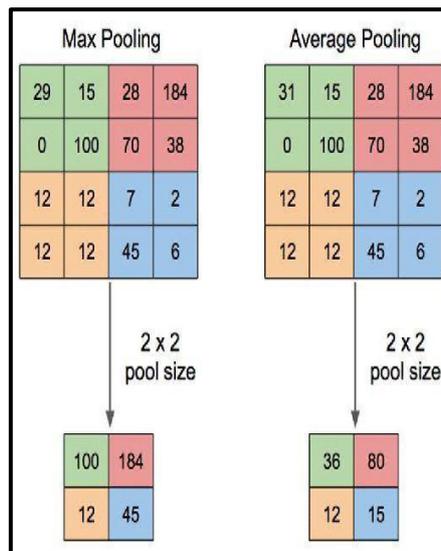
Fuente: (Llano Carmona, 2021)

Las capas de agrupación o también llamadas *pooling layers* son utilizadas para reducir las dimensiones del alto y el ancho de la entrada de la siguiente capa convolucional. Es utilizada para la reducción de sobrecarga de cálculo ya que al utilizar muchas capas las redes neuronales convolucionales el coste computacional es muy elevado para el procesamiento de los parámetros y esto nos conlleva a reducir el tamaño, aunque vamos a perder información, pero nos ayudará para realizar el sobreajuste. (Llano Carmona, 2021)

Entre las principales capas de agrupación tenemos al *max pooling* y *average pooling*.

Figura 7

Max Pooling y Average Pooling

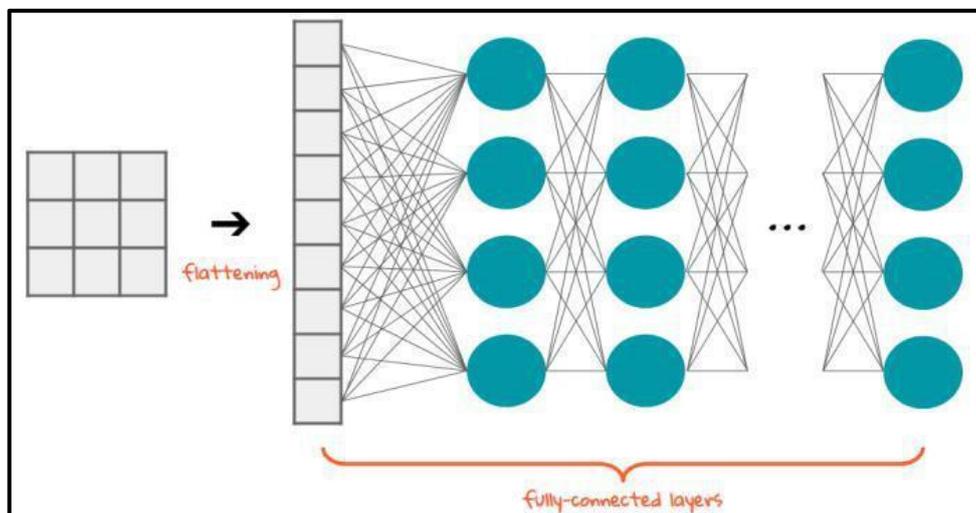


Fuente: (Llano Carmona, 2021)

Las capas totalmente conectadas o *fully-connected layers* conectan a una sola matriz en una sola dimensión llamada *flatten*, que es conectada con la capa de salida que tendrá las neuronas con las clases que estamos clasificando en la cual se va a predecir la etiqueta correcta.

Figura 8

Capa Flatten



Fuente: (Llano Carmona, 2021)

2.2.14 TensorFlow

Es una biblioteca que ha sido desarrollada por Google Brain y se utiliza para las mismas aplicaciones de aprendizaje automático de Google y para redes neuronales profundas.

2.2.15 Keras

Está escrita en el lenguaje de programación de Python, es una de las API de redes neuronales de alto nivel, su principal motor de back-end es tensorflow, se estructura en dos tipos de modelos los cuales son modelo Sequential y la clase Model.

2.2.16 Aplicaciones móviles

Las aplicaciones móviles han sido desarrolladas para ser ejecutadas en dispositivos inteligentes, como las tabletas, smartphones, otros dispositivos móviles, en la cual permite interactuar a los usuarios con los diferentes conjuntos de tareas facilitando las tareas a desarrollar.

Está distribuido en las diferentes plataformas entre ellas tenemos Android, iOS, etc.

2.2.17 Lenguaje para desarrollo en Android

2.2.17.1 Java

Es un lenguaje de programación orientada a objetos que cuenta con una gran comunidad de desarrolladores, así como una gran cantidad de librerías que ayudan a cumplir diferentes funciones.

Es el lenguaje de programación más utilizados ya que tiene compatibilidad con muchas plataformas.

2.2.17.2 Kotlin

Es un lenguaje de programación usado para el desarrollo de aplicaciones móviles, facilita el manejo de código y la reducción de las líneas de código, la cual permite la inclusión de este lenguaje por la compatibilidad con java la cual permite la reutilización de código.

2.2.17.3 Flutter

Es un framework que se utiliza para diseñar interfaces gráficas con mucha calidad, es usado para el desarrollo de aplicaciones móviles con sistemas operativos iOS y Android, la cual ayuda a realizar de forma más rápida y precisa las interfaces gráficas.

2.2.17.4 Firebase

Firebase es una plataforma de aplicaciones web, ayuda a los programadores a crear aplicaciones de alta calidad.

2.2.17.5 Firebase para Android

“Los desarrolladores pueden iniciar pruebas en la infraestructura de la nube que es utilizado para probar las aplicaciones, la cual acelera el desarrollo de las aplicaciones teniendo una infraestructura gestionada” (Chunnu Khawas & Pritam Shah , 2018)

CAPÍTULO III: MARCO METODOLÓGICO

3.1 Tipo de investigación

- **Según su alcance**, Correlacional, porque lo que pretende esta investigación es medir el grado de relación y la interacción de las variables entre sí.
- **Según su propósito**, Aplicada, porque para la implementación de esta aplicación se ha usado redes neuronales convolucionales con el objetivo de apoyar a los agricultores en la detección de enfermedades de sus plantaciones de café.
- **Según la contrastación de hipótesis**, la investigación es experimental, porque se va a establecer una relación causal con el fin de evaluar los efectos que se manifiestan entre nuestra variable dependiente e independiente.
- **Según la naturaleza de la información**, Cuantitativa, ya que la información y los datos recogidos serán tabulados y evaluados con estadística descriptiva.

3.2 Variables

3.2.1 *Variable dependiente*

VD: Detección de enfermedades en las plantas de café

3.2.2 *Variable independiente*

VI: Aplicación móvil basada en redes neuronales convolucionales.

3.3 Tabla de operacionalización de variable.

Tabla 11

Tabla de operacionalización de Variable

Variable	Definición	Dimensión	Indicadores
Detección de enfermedades en las plantas de café (VD).	“Dificultad que presentan los agricultores para detectar y reconocer enfermedades que presentan las plantas de café” (Kumar, Gupta, Madhav, & Sachin, 2020).	Interacción del agricultor con las plantas de café.	Número de enfermedades reconocidas por los agricultores.
			Costo en la identificación de las enfermedades de café.
			Cantidad promedio de agroquímicos usados correctamente en las plantas de café.
Aplicación móvil basada en redes neuronales convolucionales (VI).	“Sistema de detección que ayudará en el diagnóstico de las enfermedades demostrando su eficacia y precisión en la clasificación de imágenes y reconocimiento de patrones” (Kumar, Gupta, Madhav, & Sachin, 2020).	Fiabilidad	Tiempo promedio en la identificación de enfermedades en las plantas de café.
			Número de aciertos en detectar las enfermedades en las plantas de café.
			Número de aciertos en el tipo de agroquímicos a utilizar correctamente.

Fuente: Elaboración propia

3.4 Población y muestra

3.4.1 Población

La recolección de datos para la presente investigación se realizó en 1 hectárea de café en la cual contiene un total de 4012 plantas de café, se desarrolló en el distrito de Chirinos, provincia San Ignacio, departamento de Cajamarca.

3.4.2 Muestra

$$n = \frac{N * z^2 * p * q}{d^2 * (N - 1) + Z^2 * p * q}$$

Donde:

n: Tamaño de la muestra

N: Tamaño de la población

Z: Valor crítico de Z, si la seguridad es 95% (en este caso 1.96)

p: Proporción Aproximada (en este caso 5% =0.05)

q: Proporción de referencia, 1-p (en este caso 1 – 0.05 =0.95)

d: Nivel de Precisión absoluta 5% (en este caso seria 0.05)

Fuente: (Aguilar-Barojas, 2005)

$$n = \frac{N * z^2 * p * q}{d^2 * (N - 1) + Z^2 * p * q}$$

$$n = \frac{4012 * 1.96^2 * 0.05 * 0.95}{0.05^2 * (4012 - 1) + 1.96^2 * 0.05 * 0.95}$$

$$n = 72$$

Para realizar la medición de las prepruebas de un total de 4012 plantas de café, se tomará como muestra a 72 plantas de café en un muestreo aleatorio para hacer los respectivos análisis de la presente investigación.

3.5 Técnicas, instrumentos de recolección de datos

Tabla 12

Técnicas, Instrumentos de Recolección de Datos

TÉCNICA	INSTRUMENTO	PROPÓSITO
Entrevista	Formato de entrevista (Ver anexo-02)	Conocer a profundidad los tipos de enfermedades que atacan a las plantas de café.
Encuesta	Formato de encuesta (Ver anexo-03)	Recolectar información sobre los dispositivos móviles que usan los agricultores, así como el sistema operativo de sus Smartphones.

Fuente: Elaboración Propia

3.6 Diseño de contrastación de hipótesis

Tabla 13

Contrastación de Hipótesis

PRETEST	INSTRUMENTO	POSTEST
Se analizaron las 72 imágenes aleatorias tomadas con un experto agrónomo, para determinar la identificación correcta de las enfermedades en las plantas de café.	Aplicación móvil basada en redes neuronales convolucionales.	Se evaluará la fiabilidad y la precisión de la cantidad de enfermedades detectadas correctamente con la aplicación móvil, de las mismas 72 muestras tomadas aleatoriamente.

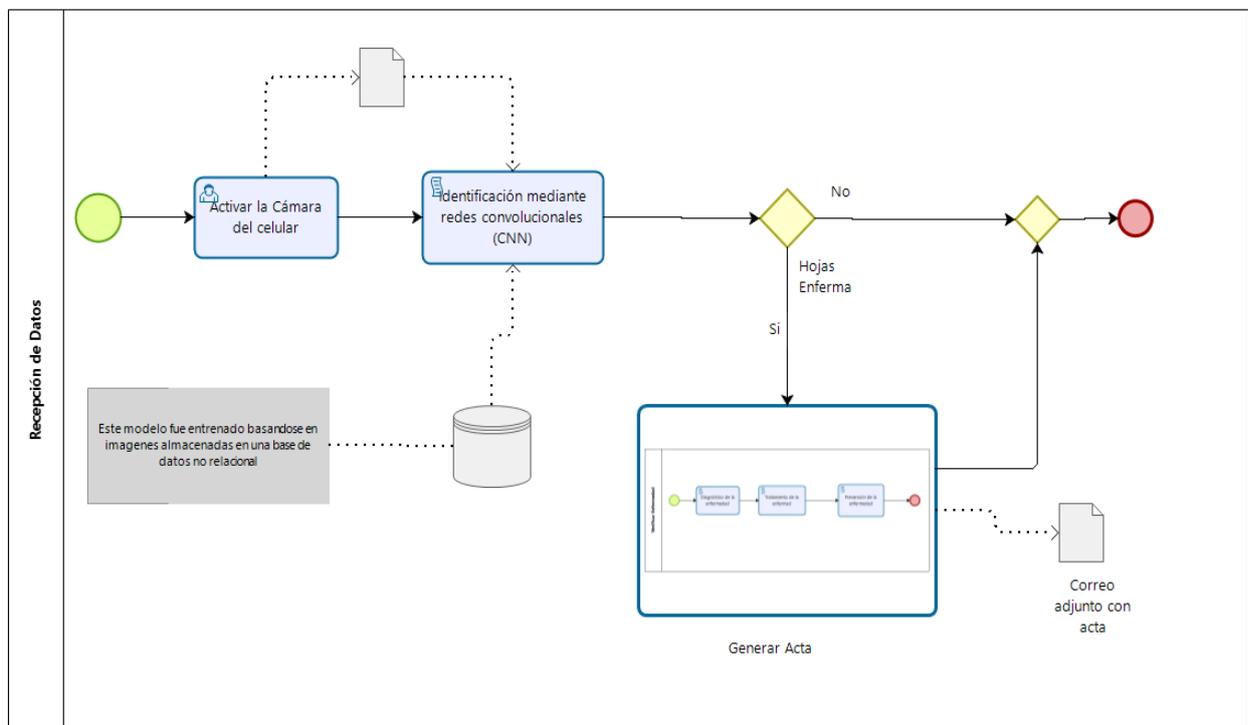
Fuente: Elaboración propia

3.7 Metodología

La metodología de este trabajo de investigación está basada en que el usuario (Agricultor) al abrir la aplicación móvil activará la cámara de manera automática. Las imágenes capturadas con el aplicativo pasarán por un proceso de identificación de redes neuronales convolucionales (CNN), si en caso las hojas de las plantas de café presentan enfermedades se genera un reporte de los diferentes agroquímicos que deben utilizarse para tratar las plantas.

Figura 9

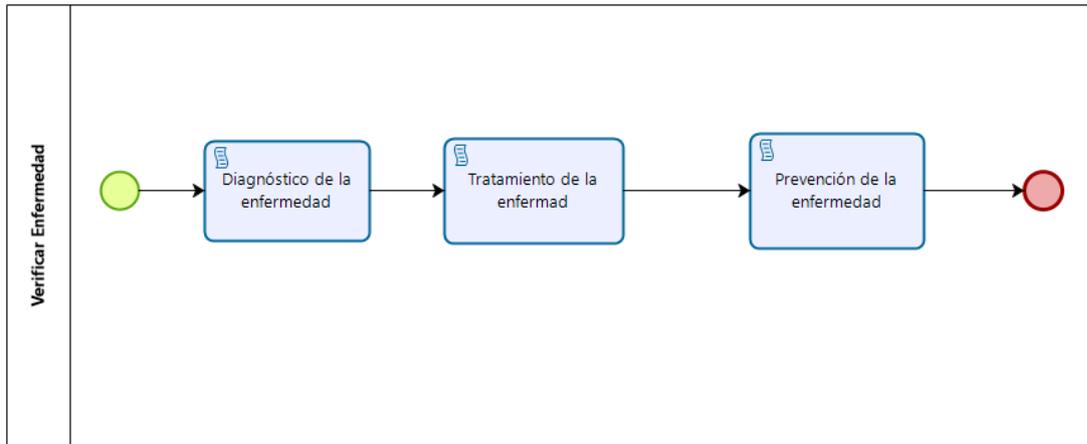
Metodología de la Investigación



Fuente: Elaboración Propia

Figura 10

Metodología de la Investigación



Fuente: Elaboración Propia

CAPÍTULO IV: DESARROLLO

4.1 Generalidades del distrito de Chirinos

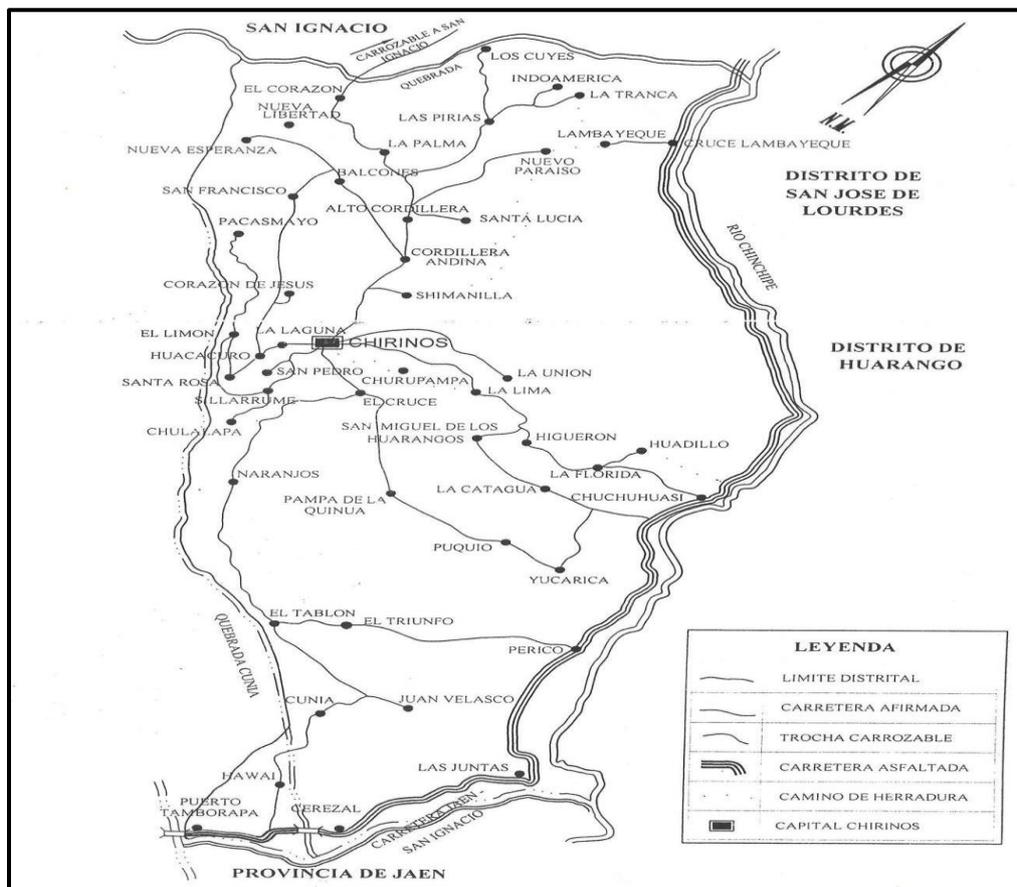
4.1.1 Ubicación Geográfica

El distrito de Chirinos, provincia de san Ignacio, departamento de Cajamarca está ubicado a una altitud de 1858 m.s.n.m, en la zona de la ceja de la selva, teniendo sus límites por el norte con el distrito de san Ignacio, por el sur con la provincia de jaén con el este con el distrito de Huarango y por el oeste con el distrito de la Coipa.

Teniendo un clima muy variado siendo un clima frío en las alturas y caluroso en las partes bajas.

Figura 11

Ubicación Geográfica del Distrito de Chirinos



Fuente: (Municipalidad Distrital Chirinos, 2022)

4.2 Requisitos y preparación

4.2.1 Requisitos previos

La presente investigación se desarrolló en la plataforma de código abierto Kaggle con un lenguaje de programación de Python en su versión 3.7.12 y TensorFlow en su versión de 2.6.3.

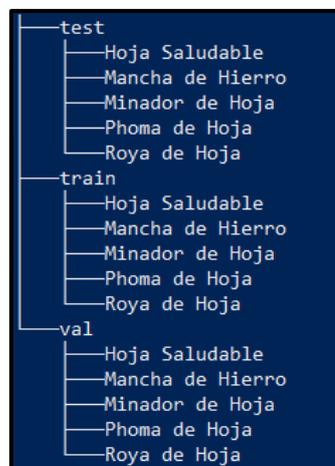
Para el entrenamiento de este modelo se adquirió una laptop con un procesador Core I7 10th Gen y una tarjeta gráfica Nvidia GeForce MX230, además, de un celular con Android 9.0 para el procesamiento de las imágenes.

4.2.2 Preparación del entorno de trabajo

Para el desarrollo de la presente investigación, en primer lugar, se utilizó el lenguaje de programación Python, y la plataforma de desarrollo Kaggle, en la cual se usó la siguiente estructura de directorios:

Figura 12

Directorios de Trabajo



Fuente: Elaboración Propia

El directorio data del proyecto se dividió en 3 sub directorios como punto de inicio de la aplicación, en las cuales son test, train, val, el sub directorio train (entrenamiento), el sub directorio val (validación), y el sub directorio test (pruebas), tienen data de las 5 enfermedades que afectan a las hojas de café, siendo sub divididas en carpetas que tienen los siguientes nombres: Hoja Saludable, Mancha de Hierro, Minador de Hoja, Phoma de Hoja y Roya de Hoja de Hoja.

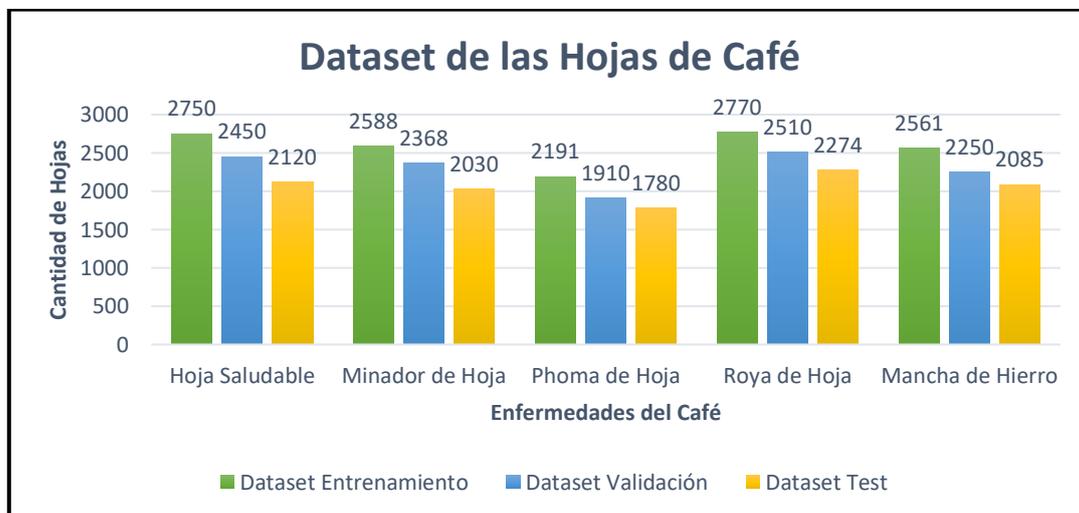
4.3 Procesamiento de datos

Los datos procesados contienen una gran variedad de imágenes de hoja de café para la detección de las enfermedades. En el preprocesamiento de los datos se han eliminado detalles no deseados y se han mejorado características de los síntomas de las hojas del café.

El conjunto de datos se divide en tres partes: la primera parte dataset Entrenamiento, contiene imágenes originales de las hojas y la segunda parte, dataset Validación, contiene imágenes recortadas de los síntomas de las hojas y por último las imágenes de los Test.

Figura 13

Dataset de las Hojas del Café



Fuente: Elaboración Propia

4.4 Descripción de los conjuntos de imágenes

El conjunto total de datos contiene 34,637 imágenes que son divididas entre las 5 diferentes clases de hojas entre ellas tenemos: Hoja saludable, minador de hoja, phoma de hoja, roya de hoja, mancha de hierro.

El conjunto de hojas para entrenamiento hay un total de 12,860 imágenes que son recortadas en sus partes asintomáticas de las hojas que estas etiquetadas según las clases de las hojas entre ellas tenemos: Hoja saludable, minador de hoja, phoma de hoja, roja de hoja, mancha de hierro, todas las imágenes tienen unas dimensiones de 128x128 pixeles, asimismo el conjunto de datos de validación contiene un total de 11,488 imágenes para comprobar el modelo generado con los datos de entrenamiento, finalmente para test contiene un total de 10,289 imágenes que serán utilizados para pruebas.

4.5 Funcionamiento de la red

La red propuesta para esta investigación es utilizar un modelo secuencial para la clasificación, el presente modelo ha sido implementado para usar la librería Keras en lenguaje Python.

El algoritmo de optimización utilizado en la presente investigación es el algoritmo ADAM, su rendimiento a comparación a otros algoritmos de optimización de descenso de gradiente convencionales es mucho más rápido para actualizar los parámetros, las imágenes proporcionadas para el entrenamiento son introducidas en lotes de 32 y un valor de épocas de

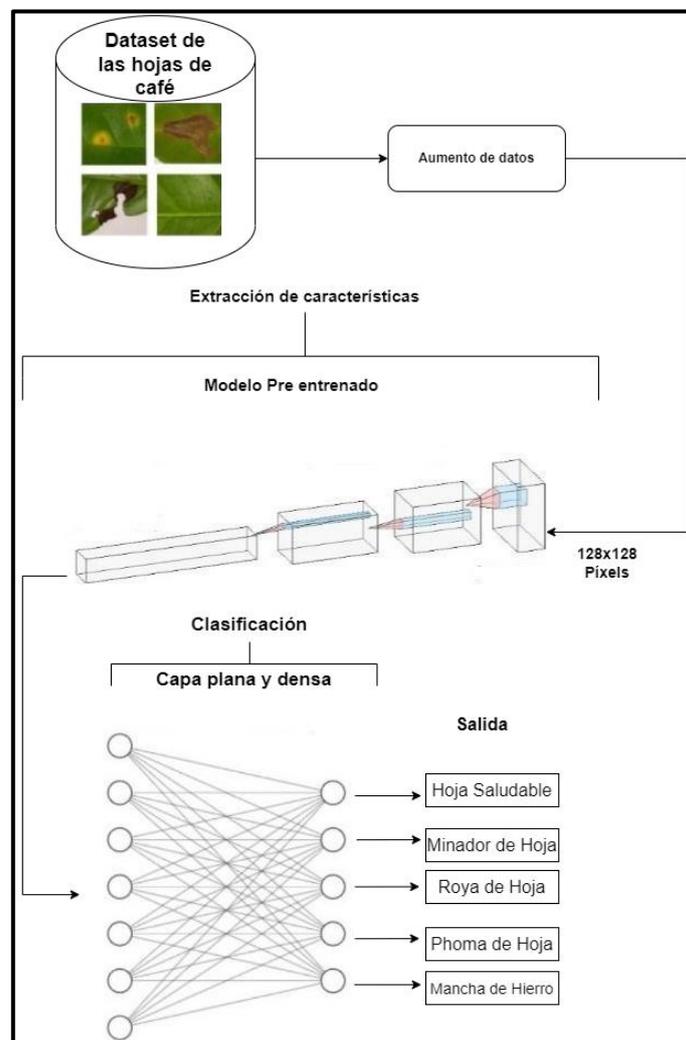
20, para el entrenamiento se utilizan imágenes de un tamaño de 128x128 píxeles pertenecientes a las 5 clases diferentes.

4.5.1 Metodología de la red neuronal convolucional

Esta sección describe el modelo que se va a utilizar para la clasificación de las clases pertenecientes a nuestro conjunto de datos, explicaremos sus arquitecturas adentrándonos en las distintas capas, entrenamiento, evaluación y predicción.

Figura 14

Metodología de la Red Neuronal Convolucional



Fuente: (Kumar, Gupta, Madhav, & Sachin, 2020)

4.5.2 Data augmentation

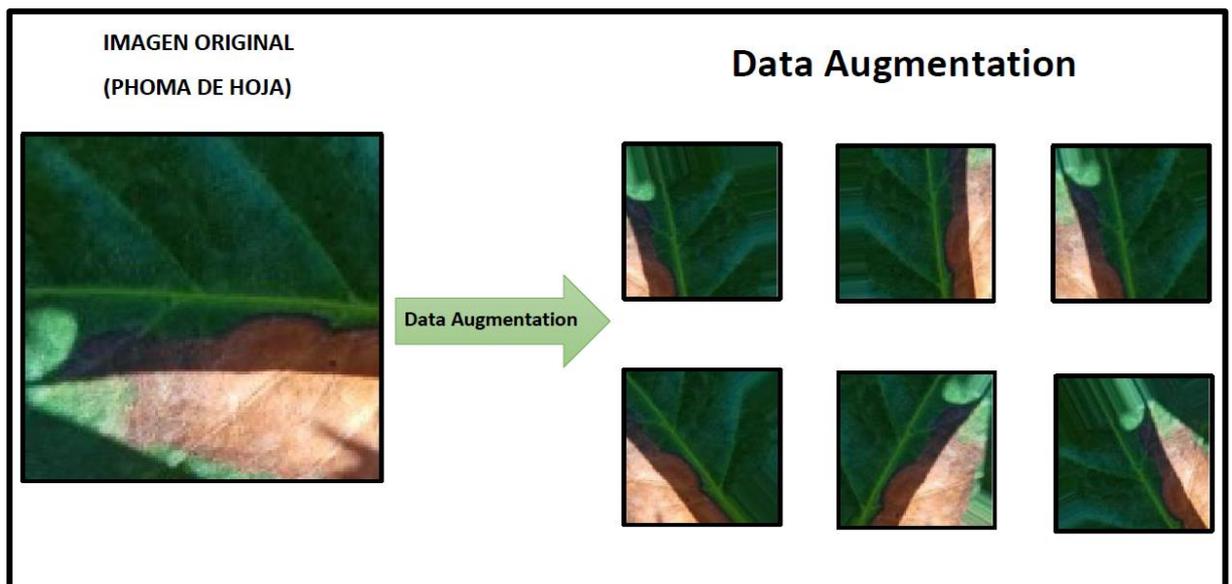
Es una técnica que permite aumentar el dataset, en caso tengamos una cantidad pequeña de imágenes. Data augmentation se utilizó para mejorar el sobreajuste y el aprendizaje de nuestra red neuronal.

Se usó esta técnica porque proporciona muchas formas de mejorar las imágenes como lo son: girar la imagen original, cambiar las condiciones de iluminación, recortarla de forma diferente de modo que la imagen pueda generar diferentes submuestras.

La biblioteca Keras, proporciona una clase llamada *ImageDataGenerator*, que se encarga de aplicar transformaciones espaciales a las imágenes.

Figura 15

Data Augmentation



Fuente: Elaboración Propia

4.6 Modelo

Este modelo está basado en una arquitectura en la cual veremos un mayor número de capas convolucionales y se implementó la técnica de *Early stopping*.

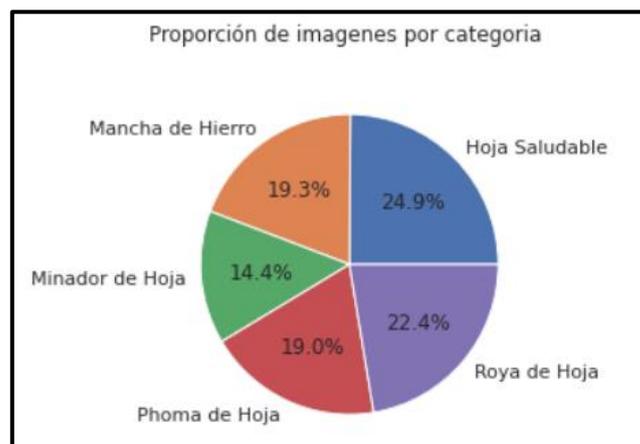
4.6.1 Datos

Para la importación de los datos del dataset, se utilizó la clase que proporciona Keras llamada *ImageDataGenerator*, en donde se va aprovechar para reescalar los píxeles entre los valores 0 y 1. Dicha clase ofrece un método llamado *Flow-from-directory* a los que vamos a pasar los siguientes parámetros: la ruta de nuestro directorio, el tamaño de nuestras imágenes (128x128), los canales de color RGB (3 canales), el tamaño de los lotes o “*batch size*” (32) y el tipo de clase (“*categorical*”).

En la clase *categorical* se aplica en el dataset el llamado “*one-hot encoding*”, que asigna la etiqueta correspondiente a cada imagen en lugar de un valor numérico. Este proceso se llevó a cabo tanto en las imágenes de entrenamiento como en las imágenes de validación y test.

Figura 16

Clasificación de las Enfermedades de la Hoja del café por categoría



Fuente: Elaboración Propia

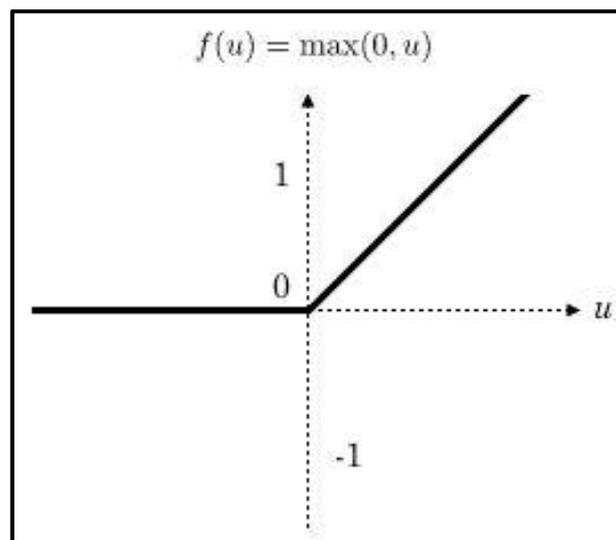
4.6.2 Arquitectura

El modelo que se usó es “*Sequential*”, ya que permite construir el modelo capa por capa, para agregar las capas se utilizó la función “*add*”, pasando como parámetro el tipo de capa que queremos agregar y sus distintas características.

Se dividió la estructura en 7 bloques, el primer bloque está constituido por 2 capas de tipo *conv2D* que va a tener los siguientes parámetros: 32 filtros con ventanas de 3x3 y la función de activación *Relu* para ambas convoluciones seguido de una capa de *MaxPooling* con un tamaño de ventana 2x2.

Figura 17

Función Relu



Fuente: (Llano Carmona, 2021)

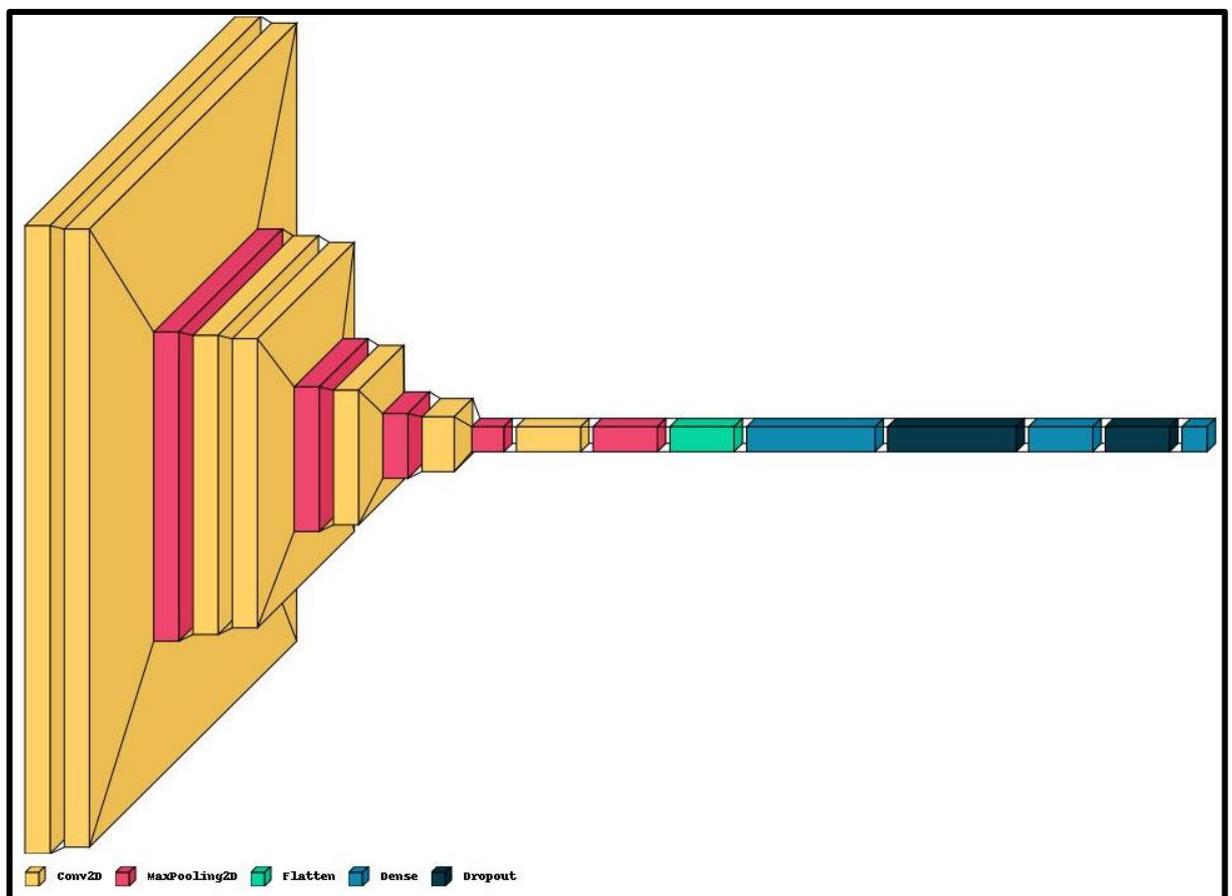
El segundo bloque es similar al primer bloque, el número de filtros queda multiplicado por 2 con un total de 64 filtros, con esta misma relación, se formaron los tres bloques siguientes,

compuesto de pares de capas de *conv2D* y *MaxPooling*, multiplicando en cada bloque el número de filtros por 2.

Los dos últimos bloques conforman el aplanamiento de la matriz en una sola dimensión, quiere decir con una capa *Flatten* y la salida se obtuvo una red compuesta por 3 capas de tipo *Dense* interferidas por las capas de *Dropout*.

Figura 18

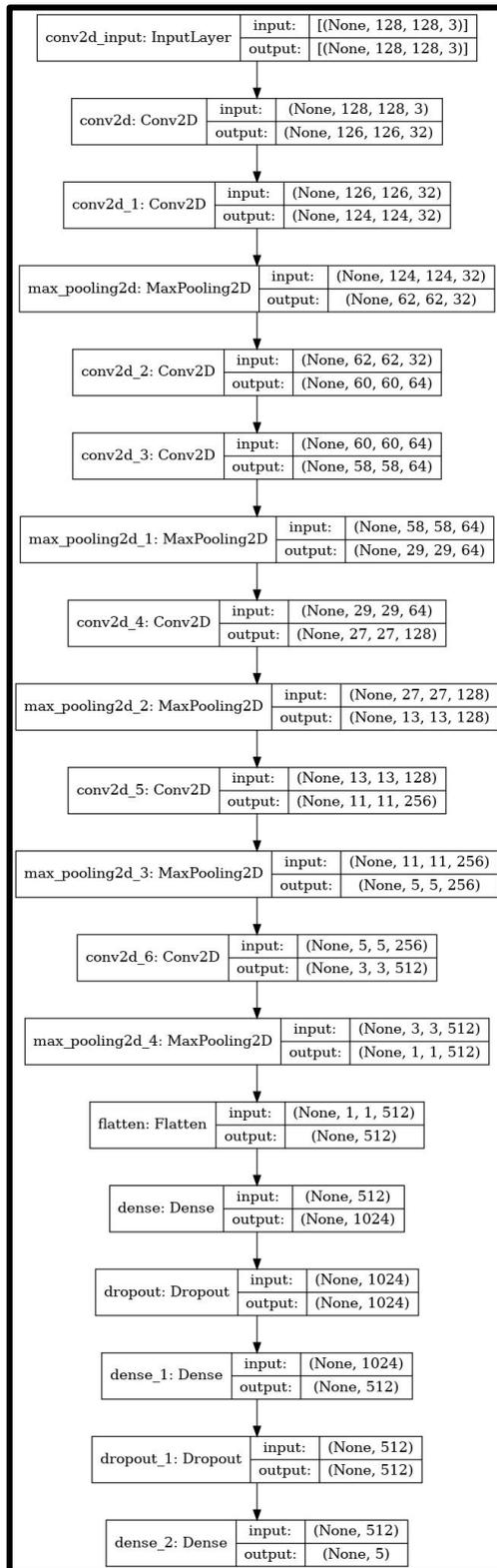
Arquitectura por capas



Fuente: Elaboración Propia

Figura 19

Capas del Modelo



Fuente: Elaboración Propia

Figura 20

Modelo Secuencial

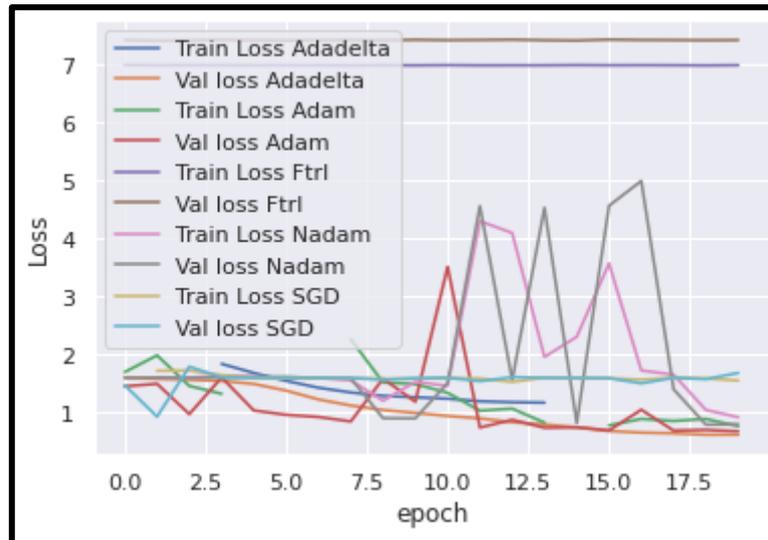
```
Model: "sequential_1"
Layer (type)                Output Shape                Param #
-----
conv2d_7 (Conv2D)           (None, 126, 126, 32)       896
conv2d_8 (Conv2D)           (None, 124, 124, 32)       9248
max_pooling2d_5 (MaxPooling2 (None, 62, 62, 32)         0
conv2d_9 (Conv2D)           (None, 60, 60, 64)         18496
conv2d_10 (Conv2D)          (None, 58, 58, 64)         36928
max_pooling2d_6 (MaxPooling2 (None, 29, 29, 64)         0
conv2d_11 (Conv2D)          (None, 27, 27, 128)        73856
max_pooling2d_7 (MaxPooling2 (None, 13, 13, 128)        0
conv2d_12 (Conv2D)          (None, 11, 11, 256)        295168
max_pooling2d_8 (MaxPooling2 (None, 5, 5, 256)         0
conv2d_13 (Conv2D)          (None, 3, 3, 512)          1180160
max_pooling2d_9 (MaxPooling2 (None, 1, 1, 512)         0
flatten_1 (Flatten)         (None, 512)                 0
dense_3 (Dense)             (None, 1024)                525312
dropout_2 (Dropout)         (None, 1024)                0
dense_4 (Dense)             (None, 512)                 524800
dropout_3 (Dropout)         (None, 512)                 0
dense_5 (Dense)             (None, 5)                   2565
-----
Total params: 2,667,429
Trainable params: 2,667,429
Non-trainable params: 0
```

Fuente: Elaboración Propia

Finalmente, compilamos el modelo donde se utilizó como optimizador el algoritmo ADAM, este algoritmo nos va a optimizar los valores de los parámetros para reducir el error cometido por la red bajando la tasa de aprendizaje del optimizador Adam a 0.0001, tomando el valor por defecto 0.001.

Figura 21

Algoritmos de Optimización



Fuente: Elaboración Propia

El siguiente parámetro que se utilizó es la función de coste, que va a determinar el error entre el valor estimado y el valor real, con lo cual se va a optimizar los parámetros de la red neuronal. Se eligió el parámetro de *“categorical_crossentropy”*, que aumenta a medida que la probabilidad predicha diverge de la etiqueta real.

Finalmente, nuestro objetivo fue minimizar el valor de la función de coste, para conseguir una mayor predicción en nuestro clasificador de imágenes, por lo tanto, para medir la métrica de nuestro modelo elegimos *“accuracy”*, esta métrica mide la precisión de nuestro modelo.

Figura 22

Métrica de Precisión

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

Fuente: (Llano Carmona, 2021)

4.6.3 Entrenamiento

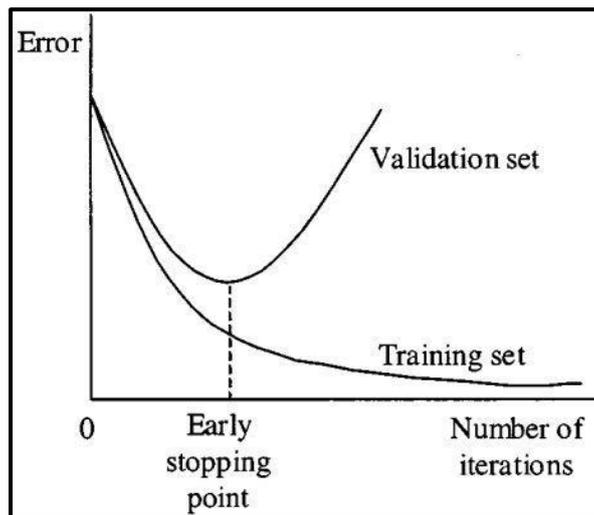
La biblioteca Keras tiene una función llamada *Fit*, la cual permitió entrenar a nuestro modelo con los siguientes parámetros:

Primeramente, se agregó un generador de imágenes de entrenamiento, seguido con el parámetro llamado “steps_per_epoch”, la cual ayudo con el número de pasos por lotes (“batch size”) en cada época, seguido agregamos un generador de imágenes de validación con su respectivo parámetro llamado "validation steps" equivalente a "steps_per_epoch", y finalmente el parámetro de "epochs", que nos indicó el número de pasadas de todo el conjunto de entrenamiento la cual lo asignamos el valor de 20.

Por lo tanto, se agregó la técnica de regularización llamada “Early Stopping”, esta técnica permite entrenar grandes épocas de entrenamiento y se detiene una vez que el modelo del conjunto de datos visualiza que el rendimiento del entrenamiento no mejora la ganancia, en nuestro caso hemos monitorizado el valor de “val_loss”, agregando un parámetro de “patience” con un valor de 3, esto quiere decir que debe haber alguna mejora en nuestros indicadores en 3 épocas.

Figura 23

Early Stopping



Fuente: (Llano Carmona, 2021)

Inicializamos el entrenamiento con 20 épocas, el modelo finalizó el aprendizaje en la época 9 gracias a la técnica Early Stopping, tras comprobar que nuestro parámetro de monitorizar no disminuye tras las "epochs" 7, 8 y 9.

Figura 24

Entrenamiento con 20 Épocas

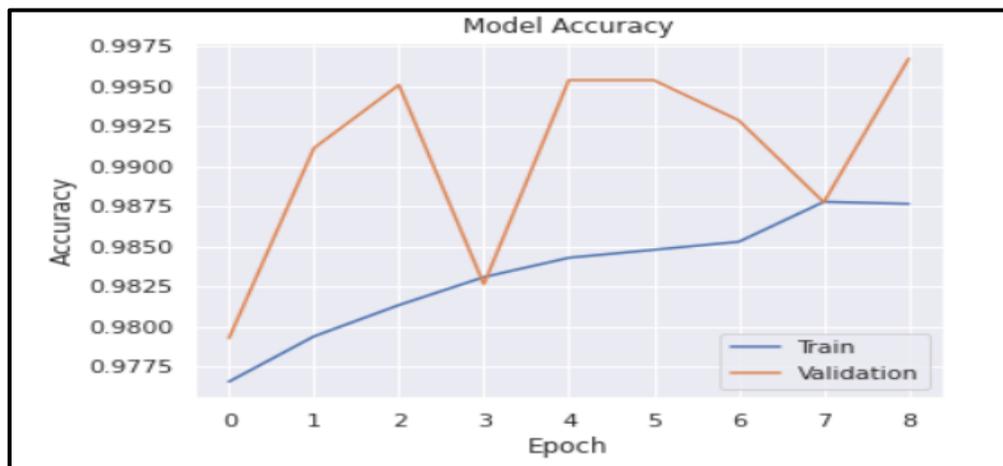
```
Epoch 1/20
752/752 [=====] - 133s 176ms/step - loss: 0.0678 - accuracy: 0.9766 - val_loss: 0.0681 - val_accuracy: 0.9793
Epoch 2/20
752/752 [=====] - 134s 178ms/step - loss: 0.0608 - accuracy: 0.9794 - val_loss: 0.0263 - val_accuracy: 0.9911
Epoch 3/20
752/752 [=====] - 132s 175ms/step - loss: 0.0567 - accuracy: 0.9813 - val_loss: 0.0189 - val_accuracy: 0.9951
Epoch 4/20
752/752 [=====] - 132s 175ms/step - loss: 0.0521 - accuracy: 0.9831 - val_loss: 0.0424 - val_accuracy: 0.9826
Epoch 5/20
752/752 [=====] - 128s 170ms/step - loss: 0.0481 - accuracy: 0.9843 - val_loss: 0.0191 - val_accuracy: 0.9954
Epoch 6/20
752/752 [=====] - 132s 176ms/step - loss: 0.0450 - accuracy: 0.9848 - val_loss: 0.0107 - val_accuracy: 0.9954
Epoch 7/20
752/752 [=====] - 130s 173ms/step - loss: 0.0432 - accuracy: 0.9853 - val_loss: 0.0146 - val_accuracy: 0.9929
Epoch 8/20
752/752 [=====] - 128s 170ms/step - loss: 0.0369 - accuracy: 0.9878 - val_loss: 0.0391 - val_accuracy: 0.9878
Epoch 9/20
752/752 [=====] - 131s 174ms/step - loss: 0.0381 - accuracy: 0.9877 - val_loss: 0.0158 - val_accuracy: 0.9967
```

Fuente: Elaboración Propia

Para el proceso de entrenamiento se detalla mediante las gráficas de evolución del “accuracy” según los datos de entrenamiento y los datos de validación, mientras que en la segunda gráfica se muestra la función de coste o “loss”.

Figura 25

Precisión entre los Datos de Entrenamiento y Validación



Fuente: Elaboración Propia

Figura 26

Pérdida entre los Datos de Entrenamiento y Validación



Fuente: Elaboración Propia

Por lo tanto, según las gráficas podemos observar el famoso “overfitting”, por una parte, la precisión de los datos de entrenamiento crece durante las 9 épocas hasta llegar a un 98.77%, mientras que los datos de precisión de validación crecen hasta llegar a 99.67% y baja a partir de la tercera época, que toma valores de 98.13% y vuelve a subir hasta nuevamente bajar en la séptima época, para finalmente subir en su valor máximo.

La función de pérdida de los datos de entrenamiento disminuye linealmente hasta llegar a 0.04% y después de una época “epochs” empieza a subir y a bajar en la quinta época y luego vuelve a subir en la séptima época y baja en su mínimo en la octava época para los datos de pérdida de entrenamiento, mientras para los datos de pérdida de validación en la cuarta época tenemos un máximo de 0.042% luego tiende a bajar en la sexta época logrando una tasa de pérdida de 0.010%.

4.6.4 Evaluación

Para evaluar el modelo se realizó con la función “Evaluate”, que nos proporciona Keras. Tras evaluar el modelo de las imágenes de prueba los valores obtenidos son 99.6% de “accuracy” y un 0.016% de “loss”.

Figura 27

Evaluación con los Datos de Prueba

```
116/116 [=====] - 5s 37ms/step - loss: 0.0169 - accuracy: 0.9963  
loss: 0.016915911808609962 Accuracy: 99.62643384933472
```

Fuente: Elaboración Propia

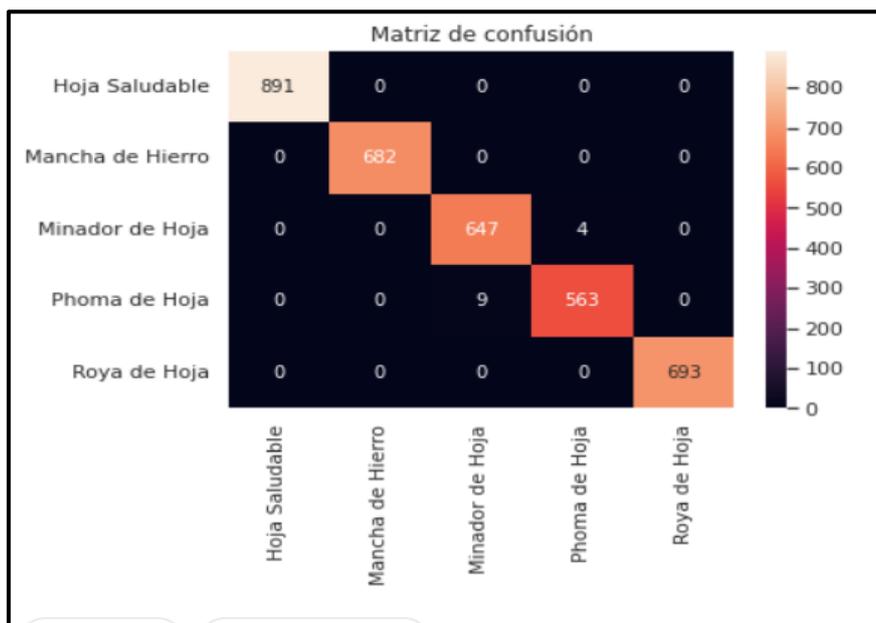
4.6.5 Predicción

Para medir la predicción de algunas imágenes, se utilizó la función “predict”, la cual vamos a pasar como parámetro una imagen preprocesada, la función “predict”, devolverá un vector de probabilidades, con valores entre 0 y 1, siendo el argumento mayor la etiqueta predicha.

Para imprimir por pantalla se usará un método para saber a qué etiqueta pertenece el índice con mayor probabilidad llamado “getcode”, al cual pasaremos como parámetro el índice indicado.

Figura 28

Matriz de Confusión



Fuente: Elaboración Propia

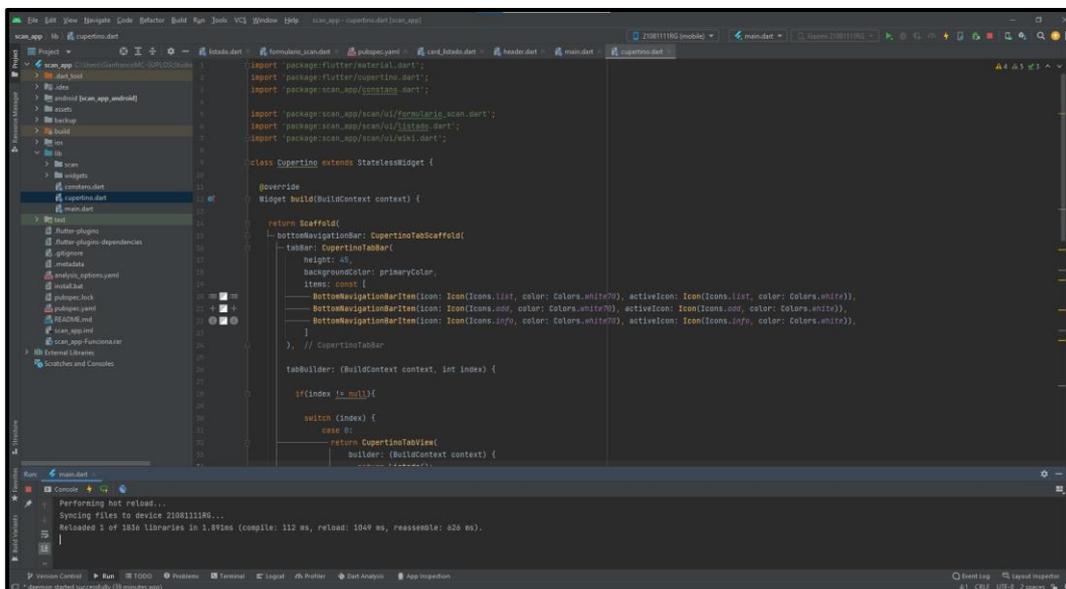
4.7 Android Studio

4.7.1 Barra de Navegación

Una de las funcionalidades que se ha utilizado para crear la aplicación móvil es la realización de una barra de navegación utilizando como framework a Flutter que permite desarrollar proyectos en Android, la barra de navegación ayudará ir a las diferentes opciones que tenga nuestra aplicación móvil.

Figura 29

Codificación de la Barra de Navegación



```
import 'package:flutter/material.dart';
import 'package:flutter/cupertino.dart';
import 'package:scan_app/cupertino.dart';

import 'package:scan_app/scan/ui/formulario_scan.dart';
import 'package:scan_app/scan/ui/listado.dart';
import 'package:scan_app/scan/ui/walk.dart';

class Cupertino extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      bottomNavigationBar: CupertinoTabScaffold(
        tabBar: CupertinoTabBar(
          height: 45,
          backgroundColor: primaryColor,
          items: const [
            BottomNavigationBarItem(icon: Icon(Icons.list, color: Colors.white70), activeIcon: Icon(Icons.list, color: Colors.white)),
            BottomNavigationBarItem(icon: Icon(Icons.add, color: Colors.white70), activeIcon: Icon(Icons.add, color: Colors.white)),
            BottomNavigationBarItem(icon: Icon(Icons.info, color: Colors.white70), activeIcon: Icon(Icons.info, color: Colors.white)),
          ]
        ), // CupertinoTabBar
        tabBuilder: (BuildContext context, int index) {
          if (index != null) {
            switch (index) {
              case 0:
                return CupertinoTabItem(
                  builder: (BuildContext context) {
                    // ...
                  }
                );
            }
          }
        }
      )
    );
  }
}
```

Fuente: Elaboración Propia

4.7.2 *Firestore Storage*

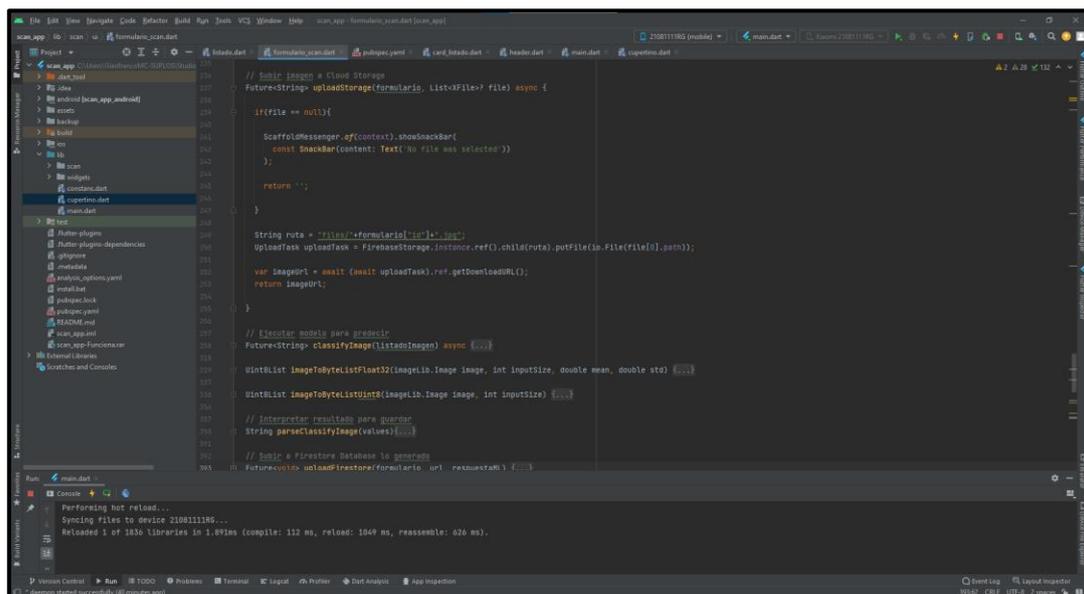
Para configurar la aplicación SCAN PLANT, primeramente, debemos conectar flutter con firebase, donde vamos a tener los datos sincronizados entre la aplicación de los agricultores en tiempo real, este modelo admite estructura de datos jerárquicos y flexibles que se organizan en colecciones.

La ruta de implementación que se ha realizado ha sido la siguiente:

- a) Integración los SDK de Cloud Firestore.
- b) Protección de datos.
- c) Agrega los datos.
- d) Obtener datos.

Figura 30

Codificación de Firestore Storage



```
226 // Subir imagen a Cloud Storage
227 Future<String> uploadStorage(formulario, List<File>? file) async {
228
229   if(file == null){
230
231     ScaffoldMessenger.of(context).showSnackBar(
232       const SnackBar(content: Text('No file was selected'))
233     );
234
235     return '';
236   }
237
238   String ruta = "fotos/${formulario['id']}_img";
239   uploadTask = FirebaseStorage.instance.ref().child(ruta).putFile(file![0].path);
240
241   var imageUrl = await (await uploadTask).ref.getDownloadURL();
242   return imageUrl;
243 }
244
245 // Escutar modelo para producir
246 Future<String> classifyImage(listadoImagen) async {
247
248   Uint8List imageToByteListFloat32(imageLib.Image image, int inputSize, double mean, double std) {
249
250   Uint8List imageToByteListUint8(imageLib.Image image, int inputSize) {
251
252   // Interpretar resultado para guardar
253   String parseClassifyImage(values) {
254
255   // Subir a Firestore database lo guardo
256   Future<void> uploadFirestore(formulario, url, imageUrl) {
```

Fuente: Elaboración Propia

4.7.3 *Firestore Database*

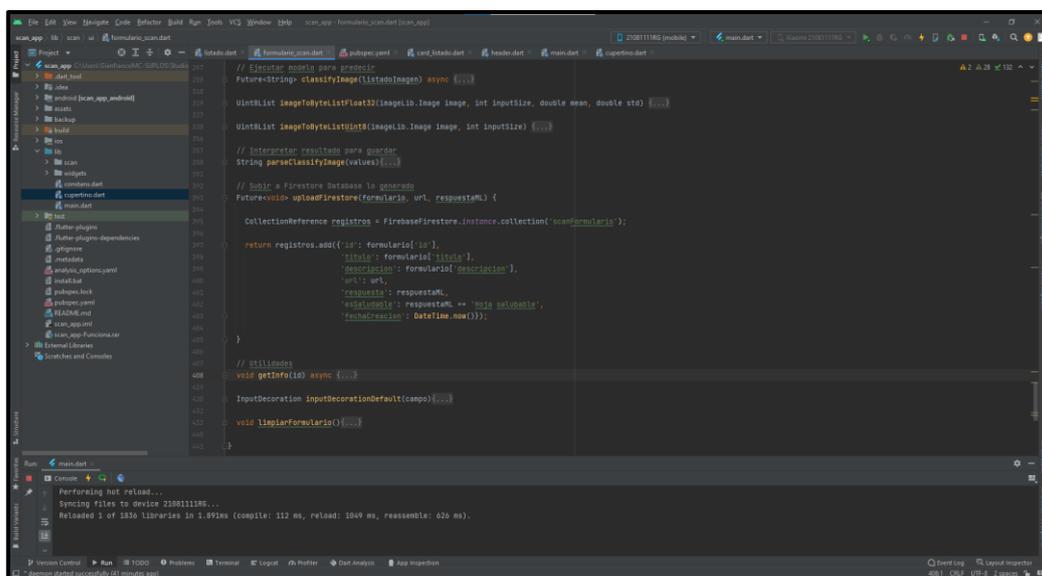
La arquitectura de los datos debe ser correctamente estructurada, como se guardarán y recuperarán los datos para que el proceso sea más fácil, almacenamos los datos como objetos JSON, no hay tablas ni registros, se agrega como un árbol de datos JSON, siendo su estructura que nos proporciona sus propias claves, como el ID de foto.

La ruta de implementación que se ha realizado ha sido la siguiente:

- a) Integración de los SDK de firebase.
- b) Creación de referencias de bases de datos en tiempo real.
- c) Establecer datos.
- d) Habilitación de persistencia sin conexión.
- e) Protección de los datos.

Figura 31

Codificación de Firestore Database



```

// Ejecutar modelo para practicar
Future<String> classifyImage(List<Image> images) async {}

Uint8List imageToByteList<T>(Image image, int inputSize) {}

// Interpretar resultado para guardar
String parseClassifyImage(values) {}

// Subir a Firestore Database lo generamos
Future<void> uploadFirestore(formulario, url, respuesta) {
  CollectionReference registros = FirebaseFirestore.instance.collection('sonido/registros');
  return registros.add({
    'id': formulario['id'],
    'titulo': formulario['titulo'],
    'descripcion': formulario['descripcion'],
    'url': url,
    'respuesta': respuesta,
    'fechaActual': respuesta['fechaActual'],
    'fechaCreacion': DateTime.now()});
}

// utilidades
void getInfo(id) async {}

InputDecoration inputDecorationDefault(campo) {}

void limpiarFormulario() {}

```

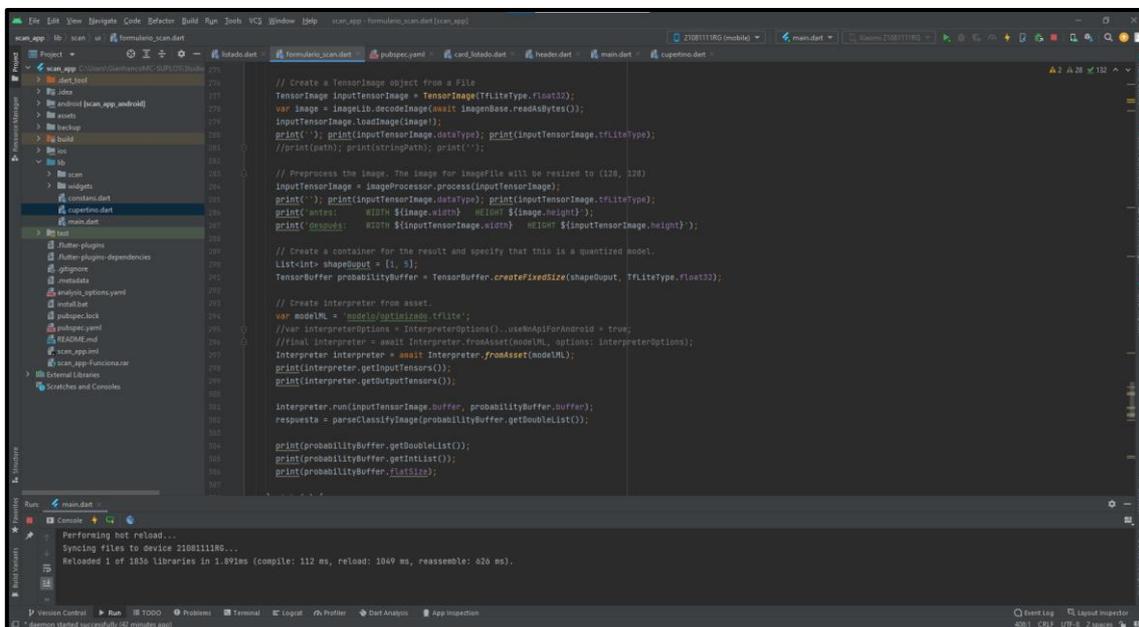
Fuente: Elaboración Propia

4.7.4 Procesamiento del Modelo Entrenado

Para el procesamiento de datos de entrada y de salida, se ha utilizado la biblioteca de TensorFlow, en la cual la Api del intérprete de TensorFlow ejecuta el modelo de aprendizaje automático en el dispositivo utilizando los tensores de forma de ByteBuffer, en la cual los datos son difíciles de manipular y depurar, siendo el objetivo final la cuantización de objetos de entrada y salida dando un resultado a una probabilidad de un punto flotante entre 0 y 1.

Figura 32

Vinculación del Api Modelo con Android



Fuente: Elaboración Propia

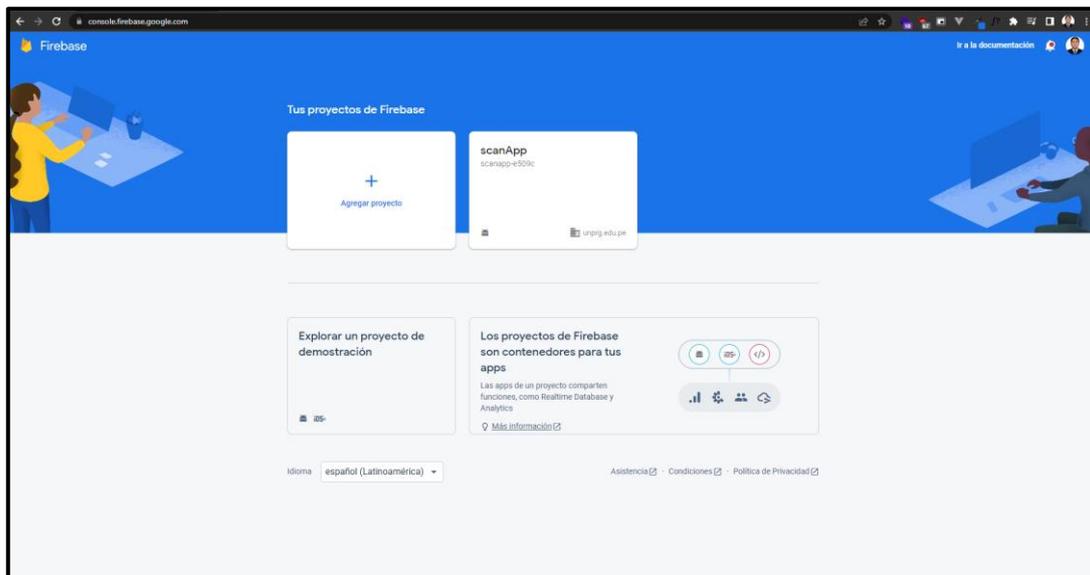
4.8 Firebase

4.8.1 Proyecto Firebase

Firebase es una plataforma en la nube que ayuda hacer más sencilla la creación de aplicaciones web como móviles, siendo que el trabajo sea muy rápido, lo primero que se ha realizado es ingresar a la consola de firebase y crear el proyecto Scan App, la cual nos permite alojar y disponer de datos e información de la aplicación en tiempo real, siendo firebase quien envía las actualizaciones que tiene la aplicación automáticamente.

Figura 33

Creación del Proyecto en Firebase



Fuente: Elaboración Propia

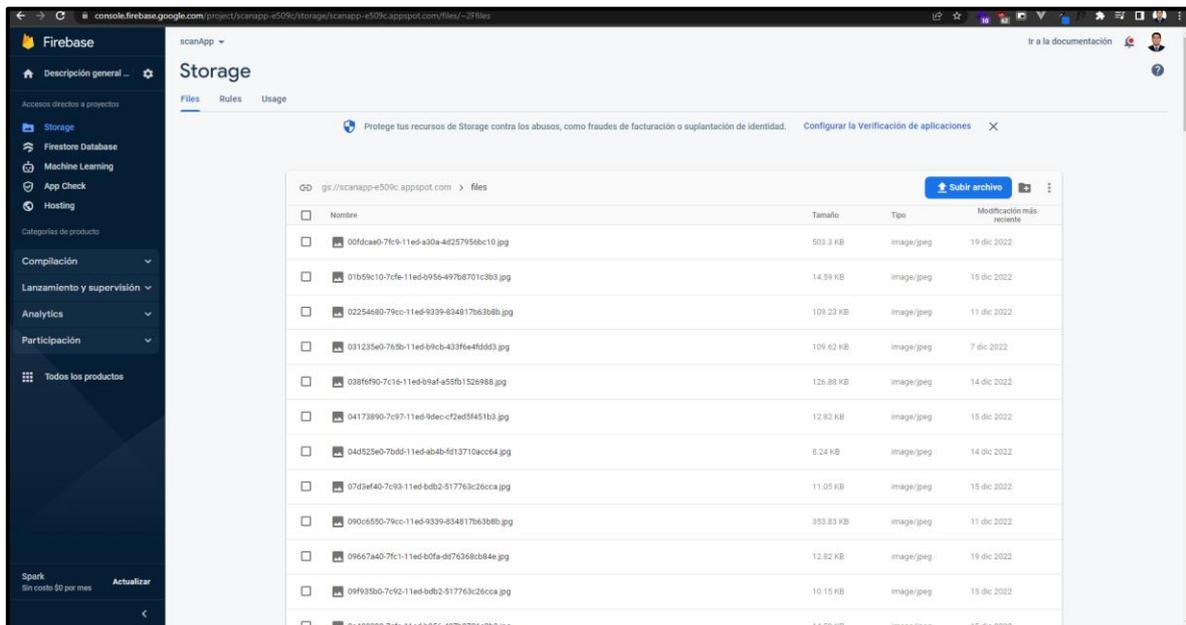
4.8.2 *Firestore Storage*

En el almacenamiento de firebase, está compuesta con un servicio de almacenamiento de todas nuestras fotos que subimos en tiempo real, principalmente la sincronización de datos es muy avanzada.

El funcionamiento de cloud Firestore, es el almacenamiento de datos en fotos que contienen campos que se asignan valores, estas fotos se almacenan en colecciones, que son contenedores de las fotos que utiliza la aplicación móvil para la organización y creación de consultas.

Figura 34

Procesamiento de Datos en Firestore Storage



Fuente: Elaboración Propia

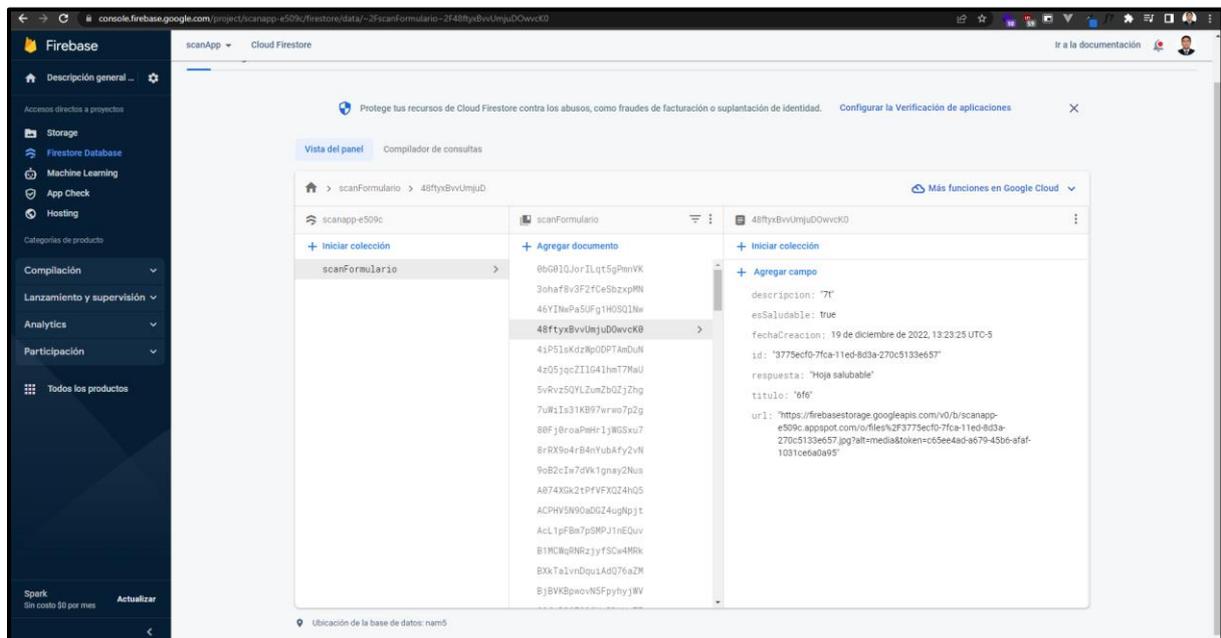
4.8.3 *Firestore Database*

El almacenamiento y la sincronización de los datos con nuestra base de datos NoSQL, ayuda a los agricultores en tiempo real a que los datos permanezcan disponibles en la aplicación.

El funcionamiento de database, permite el acceso seguro de los datos, los cuales los eventos se pueden continuar y activar en tiempo real, brindándole al usuario final una experiencia receptiva.

Figura 35

Procesamiento de Datos en Firestore Database



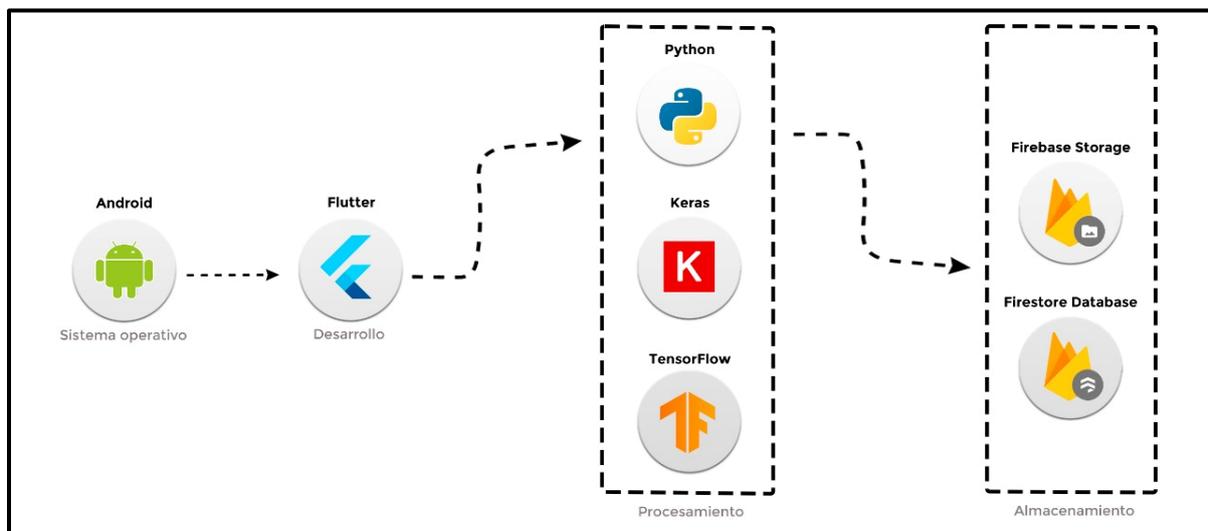
Fuente: Elaboración Propia

4.8.4 WebServices

Para la creación de la aplicación móvil SCAN PLANT, se ha realizado un webServices utilizando como sistema operativo Android, y como framework a Flutter, para el modelo se ha realizado en el lenguaje de programación Python y se ha utilizado Keras con TensorFlow como bibliotecas de software, para el almacenamiento y base de datos se ha utilizado a Firebase.

Figura 36

WebServices



Fuente: Elaboración Propia

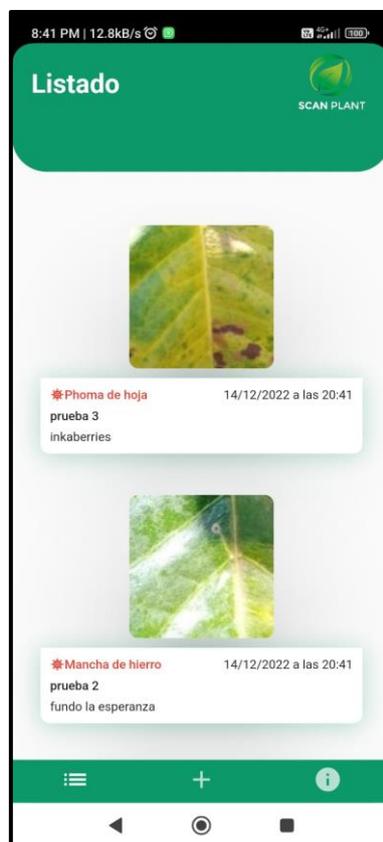
4.9 Aplicación Móvil

4.9.1 Interfaz Gráfica

La aplicación móvil se llama SCAN PLANT, es una aplicación que detecta las enfermedades que presenta las plantas de café, está disponible para Android, esta aplicación móvil el agricultor o la persona que inicia sesión, carga automáticamente nuestra interfaz de un historial de todas las enfermedades detectadas en tiempo real.

Figura 37

Interfaz Gráfica de la Aplicación Scan Plant



Fuente: Elaboración Propia

4.9.2 Formulario y análisis de datos

En nuestro formulario podemos introducir datos necesarios para poder analizar las fotos de las enfermedades de las plantas de café, cada parte tiene una opción de tener unos datos específicos de ser llenados, primeramente, llenamos el título, luego una pequeña descripción y finalmente seleccionamos la imagen que iremos a analizar.

Figura 38

Formulario de la Aplicación Scan Plant



The image shows a mobile application interface for 'Scan Plant'. At the top, there is a green header with the word 'Formulario' in white text and the 'SCAN PLANT' logo on the right. Below the header, there are two text input fields: 'Título' and 'Descripción'. Underneath these fields is a large light blue square area containing a grey icon of a picture with a mountain and a sun, representing an image selection area. At the bottom of this area, there are two blue buttons: 'Agregar' and 'Seleccionar imagen'. The bottom of the screen features a green navigation bar with three icons: a hamburger menu, a plus sign, and an information icon. Below the navigation bar is the standard Android system bar with back, home, and recent apps buttons.

Fuente: Elaboración Propia

Una vez completado el formulario, daremos en agregar, y automáticamente analizara la foto y obtendremos el resultado de la enfermedad que presenta la planta de café,

Figura 39

Análisis de Datos de la Aplicación Scan Plant



Fuente: Elaboración Propia

Finalmente, el apartado de WIKI, obtendremos un catálogo de todos los agroquímicos que los agricultores pueden utilizar para prevenir las enfermedades detectadas con la aplicación móvil.

Figura 40

Wiki de la Aplicación Scan Plant



Fuente: Elaboración Propia

CAPÍTULO V: RESULTADOS

En esta sección presentaremos una comparativa entre los resultados de nuestra aplicación móvil generados previamente con los resultados generados por el experto agrónomo.

Tabla 14

Comparativa de Resultados entre el Experto Agrónomo y la Aplicación Móvil

Nº	Técnica utilizada	Precisión (%)	Valor de pérdida
1	Experto basado en detección de enfermedades de café (Ingeniero Agrónomo)	100%	No disponible
2	Aplicación Móvil usando redes neuronales convolucionales	99.6%	0.016%

Fuente: Elaboración Propia

Tabla 15

Valores de los Hiperparámetros

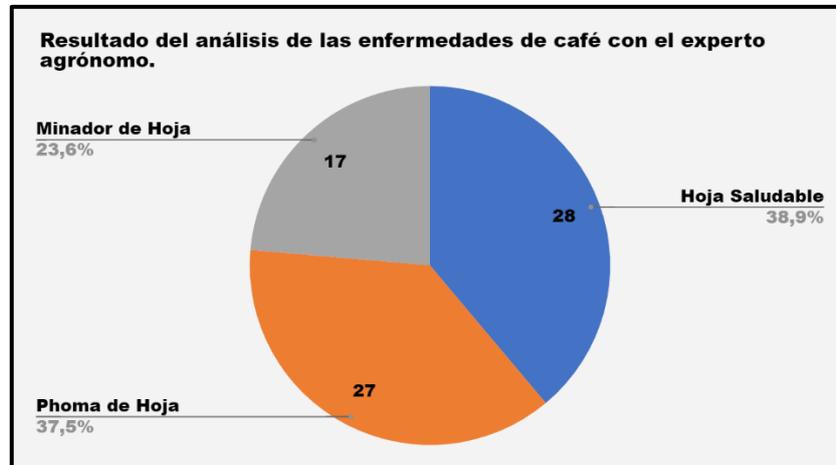
Nº	Parámetros	Valor
1	Función de activación	Relu
2	Función de pérdida	Categorical cross-entropy
3	Función de optimización	ADAM
4	Tamaño del lote	32
5	Tasa de aprendizaje	0.0001

Fuente: Elaboración Propia

Resultado del análisis del experto agrónomo usando redes neuronales convolucionales.

Figura 41

Resultado del análisis del experto agrónomo

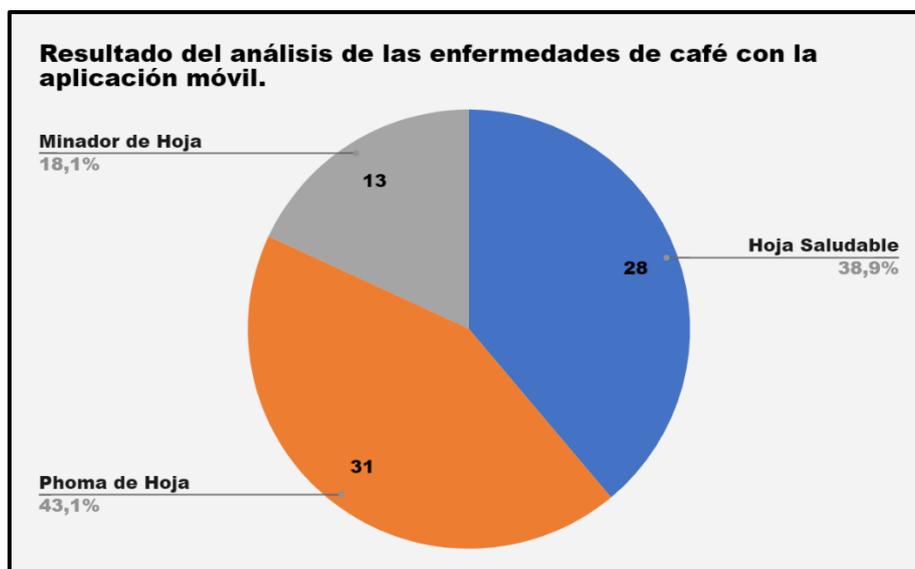


Fuente: Elaboración Propia

Resultado del análisis de la Aplicación Móvil usando redes neuronales convolucionales

Figura 42

Resultado del análisis de la Aplicación Móvil



Fuente: Elaboración Propia

Según los resultados del modelo que utiliza la aplicación móvil, la precisión de cada época aumenta y la pérdida disminuye significativamente con el aumento de las épocas, se toma como referencia a la época 6, por la técnica de Early Stopping que se detiene 3 épocas antes de la época 9 siendo la época final, por ser la época más alta y eficaz.

Tabla 16

Variación de la Precisión y del Valor de Pérdida con la Época

Número de épocas	Precisión (%)	Valor de Pérdida
1	0.979	0.068
3	0.995	0.018
6	0.992	0.0107
8	0.987	0.0391
9	0.996	0.0158

Fuente: Elaboración Propia

Se verifica en la comparativa de la muestra de las 72 imágenes, el modelo de red neuronal convolucional con la del experto agrónomo, hay diferencias respecto a los valores de los resultados del experto agrónomo con el modelo de red neuronal convolucional, es decir en la matriz de confusión del modelo de la aplicación móvil tiende a confundir las siguientes enfermedades: minador de hoja por la enfermedad phoma de hoja, en nuestra tabla 17, verificamos dichos resultados que se contrastan con la predicción del experto agrónomo.

Tabla 17

Comparativa de Resultados entre el Experto Agrónomo y la Aplicación Móvil

Enfermedad Detectada	Experto Agrónomo	Aplicación Móvil
Hoja Saludable	38.9 %	38.9%
Mancha de Hierro	-	-
Minador de Hoja	23.6 %	18.1 %
Phoma de Hoja	37.5 %	43.1 %
Roya de Hoja	-	-

Fuente: Elaboración Propia

Para validar los indicadores de la matriz de operacionalización, tenemos la técnica de la encuesta realizada a los 80 agricultores en el distrito de chirinos en el Anexo-03:

1. Para la variable dependiente: Detección de enfermedades en las plantas de café.
 - El primer indicador: número de enfermedades reconocidas por los agricultores, en la encuesta pregunta 05, hace referencia la manera que tienen agricultores para identificar las enfermedades que presenta sus plantas de café, teniendo como resultados: la observación y experiencia, método tecnológico, recurren a un experto y otro, dado que la mayoría de agricultores con un 77% tienden a identificar mediante la observación y experiencia.
 - El segundo indicador: costo en la identificación de las enfermedades de café, en la encuesta pregunta 07, hace referencia al costo que tienen los agricultores en la identificación de las enfermedades en sus plantas de café, siendo sus gastos: 30, 50, 100 soles, dado que la mayoría de agricultores con un 40% gasta aproximadamente 100 soles por planta.
 - El tercer indicador: cantidad de agroquímicos usados correctamente en las plantas de café, en la encuesta pregunta 09, hace referencia a las dosis que utilizan los agricultores para el cuidado de sus plantas, siendo un 91% de agricultores que tienen en cuenta las dosis de agroquímicos necesarios para el cuidado de sus plantas.

2. Para la variable independiente: Aplicación móvil basada en redes neuronales convolucionales.

- El primer indicador: Tiempo promedio en la identificación de enfermedades, en la encuesta pregunta 06, hace referencia que el agricultor tiene un tiempo promedio de 30 minutos en identificar la enfermedad en las plantas de café, con la aplicación móvil se logró optimizar el tiempo aproximadamente en 2 segundos para identificar la enfermedad que causa a la planta de café.
- El segundo indicador: Número de aciertos en detectar enfermedades en las plantas de café, hacemos referencia al modelo de redes neuronales convolucionales con una tasa de acierto de 99.6% y una función de pérdida de 0.016%, teniendo una matriz de confusión que tiende a confundir las enfermedades minador de hoja con la phoma de hoja.
- El tercer indicador: Número de aciertos en los tipos de agroquímicos a utilizar correctamente, hace referencia a la aplicación móvil en el menú de la wiki, se obtiene los diferentes tipos de agroquímicos que los agricultores deben utilizar para la combatir la enfermedad detectada con la aplicación móvil.

CAPÍTULO VI: DISCUSIÓN

Este trabajo de investigación estudia la integración de las redes neuronales convolucionales para la detección de enfermedades en las plantas de café, para cumplir nuestro primer objetivo es la identificación de las enfermedades que presentan las plantas de café en base a la extracción de características en las fotografías realizadas en las hojas de las plantas de café, coincide con lo que sostiene en la investigación titulado *“Aplicación móvil de detección y clasificación de “la roya” en hojas de café robusta mediante aprendizaje automático”* de (Cusme Zambrano & Loor Pinargote, 2019), en la cual hace referencia a los patrones y técnicas de reconocimiento de imágenes que determinan la identificación de las enfermedades en las plantas de café, esto acorde con el procesamiento de imágenes en nuestra aplicación móvil, siendo el procesamiento y almacenamiento de imágenes que se realizan en nuestro servidor.

Para el cumplimiento de nuestro segundo objetivo, el cual es optimizar los tiempos en la identificación y búsqueda de enfermedades en las plantas de café, se utilizó la teoría de la investigación titulada *“Disease Detection in Coffee Plants Using Convolutional Neural Network”* que tiene como autores a (Kumar, Gupta, Madhav, & Sachin, 2020), en la cual propone un modelo de red neuronal de convolución de reducción de tiempo, en la cual tiene coincidencia con nuestro modelo de entrenamiento en donde se aplicó las técnicas de aumento de datos para ampliar nuestro conjunto de dataset, logrando una alta precisión de un 99.6%.

Para el cumplimiento de nuestro tercer objetivo, el cual es determinar el tratamiento adecuado utilizando agroquímicos en las enfermedades identificadas en las plantas de café, tiene coincidencia con la investigación titulada, “Aplicación de imágenes hiperespectrales para la detección temprana de roya amarilla (*hemileya vastatrix*) en café (*coffea arábica*), en el distrito de Limbamba, provincia Rodríguez de Mendoza región Amazonas”, que tiene como autor a (López Portocarrero, 2019), en sus objetivos es determinar las infecciones tempranas en la roya amarilla en las plantas de café, por consiguiente aplica la metodología de árbol de decisión y la técnica hiperespectrales con Matlab V2010, en la que logró determinar la fiabilidad en ambos métodos, logrando así una técnica confiable de detección en las infecciones en el tejido de las hojas dando recomendaciones de qué productos agroquímicos se debería utilizarse para contrarrestar la enfermedad de la roya amarilla.

Finalmente, los datos que se obtuvieron en esta investigación aprueban la hipótesis que fue propuesta al inicio del desarrollo de la tesis según : “Implementar redes convolucionales y un sistema de inteligencia artificial, permitirá apoyar en la detección de las enfermedades en las plantas de café en el distrito de Chirinos.”, se considera verdadera, dado que se ha logrado implementar una aplicación móvil utilizando redes neuronales convolucionales para la detección de enfermedades en las plantas de café, logrando una precisión de un 99.6%.

CAPÍTULO VII: CONCLUSIONES

Este trabajo de investigación presenta el uso de redes neuronales convolucionales para la detección de las plantas de café sanas y no sanas. Las hojas no sanas de las plantas de café se clasifican en 4 clases: roya de la hoja del café, manchas hierro o Cercospora, minador de la hoja, Phoma de la hoja, que son estreses bióticos que ocurren comúnmente en las plantas de café, siendo evaluadas solamente las enfermedades de minador de la hoja, Phoma de la hoja y hoja saludable, por la muestra tomada aleatoriamente por el experto agrónomo.

Una de las principales contribuciones de nuestro trabajo es proponer una red neuronal convolucional para la detección de enfermedades en plantas de café con un nivel de precisión muy alto usando una aplicación móvil.

1. La identificación y clasificación de las enfermedades del café mediante nuestra red neuronal convolucional propuesta, tuvo una precisión del 99.6%.
2. El agricultor podrá ser ayudado con la identificación oportuna de las enfermedades de café, así mismo aumentará la producción del café en el distrito de Chirinos, Provincia de San Ignacio, Departamento de Cajamarca.

CAPÍTULO VIII: RECOMENDACIONES

Se recomienda tener fotografías de una buena calidad HD, que sean tomadas en un ambiente con buena iluminación.

Para obtener la predicción de la enfermedad de la planta de café, el agricultor debe situarse en la planta y tomar fotos de las hojas de las plantas de café que se visualicen algún tipo de enfermedad.

Para hacer una predicción aún mejor y lograr una mayor precisión para una amplia variedad de enfermedades en las plantas de café, se requieren variables adicionales como la temperatura, la humedad del aire, el momento de la plantación y el clima de la ubicación.

Como trabajo futuro, pretendemos explorar los requisitos y cambios necesarios en nuestra red neuronal para poder acomodar otras nuevas enfermedades en las plantas de café y también tener en cuenta los factores ambientales variables.

REFERENCIAS BIBLIOGRÁFICAS

- Abu Mettleq, A. S., & Abu-Naser, S. S. (2019). A Rule Based System for the Diagnosis of Coffee Diseases. *International Journal of Academic Information Systems Research (IJAIRS)*, 8.
- Aguilar-Barojas, S. (2005). *Fórmulas para el cálculo de la muestra en investigaciones de salud Salud en Tabasco*. México: Secretaría de Salud del Estado de Tabasco Villahermosa.
- Almengor, I. O. (2019). *Recomendaciones de la época para el control de la Broca del Fruto del Cafeto - Hypothenemus hampei - y el Minador de la Hoja - Leucoptera coffeella*. Asociación nacional del café.
- Amao Sulcahuaman, S. D., Callañaupa Chihuantito, H. W., Lima Ramos, A., Morocco Layme, J., Quispe Coronel, A. Y., & Zegarra Corimanya, M. A. (s.f.). *Visión Artificial*. 23.
- Balardin, R. (2010). Factors affecting fungicide efficacy in the tropics fungicides. 18.
- Chunnu Khawas, & Pritam Shah. (2018). Application of Firebase in Android App Development-A Study. *International Journal of Computer Applications*, 1-6.
- CPCC. (2021). *Cámara Peruana del Café y del Cacao*.
- Cusme Zambrano, K. D., & Loo Pinargote, A. M. (2019). *Aplicación móvil de detección y clasificación de "la roya" en hojas de café robusta mediante aprendizaje automático*. Manabí.
- Derwin, S., Wahyu, A., Miranty, L., & Muhammad, Y. (2013). Expert System in Detecting Coffee Plant Diseases. *International Journal of Electrical Energy*, 162.
- Di Donfrancesco, B., Gutierrez Guzman, N., & Chambers IV, E. (2019). Similarities and differences in sensory properties of high quality Arabica coffee in a small region of Colombia. *Food Research International*, 645-651.
- Díaz Vargas, C., & Carmen Willems, M. (2017). Línea de Base del sector café en el Perú. En *Programa de Green Commodities* (pág. 58). Biblioteca Nacional del Perú N° 2017-06972.
- El Naqa, I., & J. Murphy, M. (2015). What is Machine Learning? *Springer International Publishing Switzerland*, 1-9.
- JNC. (2020). Obtenido de <https://juntadelcafe.org.pe/estadisticas/>
- JNC. (20 de Agosto de 2021). *Exportación de café enero - diciembre*. Obtenido de Junta Nacional del Café: <https://juntadelcafe.org.pe/estadisticas/>
- JNC. (20 de Agosto de 2021). *Junta del Café*. Obtenido de EL CAFÉ DE PERÚ: <https://juntadelcafe.org.pe/el-cafe-de-peru/>
- Kumar, M., Gupta, P., Madhav, P., & Sachin. (2020). Disease Detection in Coffee Plants Using Convolutional Neural Network. *International Conference on Communication and Electronics Systems*, 1-6.

- Llano Carmona, M. d. (2021). Artificial Neurotransmitter. *dpto. Ciencias de la Computación e Inteligencia Artificial*, 1-12.
- López Portocarrero, C. A. (2019). *Aplicación de imágenes hiperespectrales para la detección temprana de roya amarilla (hemileya vastatrix) en café (coffea arábica), en el distrito de limbamba, provincia rodríguez de Mendoza región amazonas*. Chachapoyas.
- Mejía Vilet, J. R. (2005). *Procesamiento Digital de Imágenes*. San Luis Potosí: Facultad de Ingeniería UASLP.
- MIDAGRI. (20 de Agosto de 2021). Obtenido de Ministerio de Desarrollo Agrario y Riego: <https://cdn.www.gob.pe/uploads/document/file/1736968/Observatorio%20de%20los%20precios%20internacionales%20y%20nacionales%20de%20los%20commodities%2C%20enero%202021.pdf>
- Miller , S. A., Beed, F. D., & Lapaire Harmon, C. (2009). Plant disease diagnostic capabilities and networks. *Annual Review of Phytopathology*, 38.
- Minagri, M. d. (2019). *Observatorio de Commodities: Café*. Lima: Dirección de Estudios Económicos e Información Agraria (DEEIA) del Minagri.
- Mogena Soler, A. (2014). Informe de OpenCV y Tratamiento de Imágenes. 1-10.
- Municipalidad Distrital Chirinos. (2022). Obtenido de Municipalidad Distrital Chirinos: <https://www.gob.pe/munichirinos>
- Parra Plaza, J. A. (2015). Intérprete de lenguaje de signos para la comunicación de personas con discapacidad auditiva empleando procesamiento de imágenes. *Revista de Investigaciones - Universidad del Quindío*, 7.
- Piscoya Ferreñan, J. E. (2019). *Sistema de visión artificial para apoyar en la identificación de plagas y enfermedades del cultivo de sandía en el distrito de Ferreñafe*. Ferreñafe.
- Proyecto Cafe y Clima. (2017). Estudio de Mercado del Café Peruano. Aleph Soluciones Graficas.
- Roldán Ortega, B., Roshan Biswal, R., & Sánchez Delacruz, E. (2019). Detección de enfermedades en el sector agrícola. 1-9.
- Sánchez Méndez, A. G., & Arguijo Hernández, S. P. (2018). Análisis de imágenes multiespectrales para la detección. *Latindex*, 1-9.
- Zegarra M, E. (2019). *Auge agroexportador en el Perú: un análisis de sobrevivencia de productos y empresas*.

ANEXOS

ANEXO-01: CONSTANCIA DE ACEPTACIÓN DE LA EMPRESA EXPORTADORA DE CAFÉ CASMAR E.I.R.L

EXPORTADORA
DE CAFÉ CASMAR
E.I.R.L

EXPORTADORA DE
CAFÉ CASMAR
E.I.R.L

CONSTANCIA

Cordial Saludo,

Por este medio, él Gerente General, CASTILLO MARTÍNEZ, Luis, les complace notificarles la aceptación del proyecto de tesis "Aplicación Móvil usando redes neuronales convolucionales para la detección de enfermedades en las plantas de café en el distrito de Chirinos, Provincia de San Ignacio, Departamento de Cajamarca", a llevarse a cabo por los Bachilleres de la Escuela Profesional de Ingeniería de Sistemas de la Universidad Nacional Pedro Ruiz Gallo, FLORES MENDOZA, Juan Carlos y MEJIA CARHUAJULCA, Gianfranco, identificados con código universitario N° 020155635K y N° 020155640D respectivamente.

Como se indicó en la presentación del proyecto, la empresa EXPORTADORA DE CAFÉ CASMAR E.I.R.L, contribuirá con sus cultivos para detectar enfermedades en las plantas de café.

Estamos seguros de que la culminación de este proyecto se llevará a cabo bajo las condiciones y características estipuladas.

Atentamente,

EXPORTADORA DE CAFÉ
"CASMAR" E.I.R.L.

Luis Castillo Martínez
GERENTE

LUIS CASTILLO MARTÍNEZ
Gerente General

EXPORTADORA DE CAFE
CASMAR E.I.R.L
RUC: 20570802136
Cal. Yawar Fiesta Nro. 130
Sec. Pueblo Libre
Jaén

ANEXO-02: ENTREVISTA AL GERENTE GENERAL DE LA EMPRESA EXPORTADORA DE CAFÉ CASMAR E.I.R.L.



ENTREVISTA AL GERENTE GENERAL Y AL INGENIERO AGRÓNOMO DE LA EMPRESA EXPORTADORA DE CAFÉ CASMAR E.I.R.L

Nombre: Luis Castillo Martínez

Cargo: Gerente General

Empresa: EXPORTADORA DE CAFÉ CASMAR E.I.R.L

Fecha: 20-09-2022

1. ¿Cómo Gerente General de la Empresa Exportadora de Café Casmár E.I.R.L., como ayuda a los agricultores en la producción y cosecha de café?

La empresa Exportadora de Café Casmár E.I.R.L., ayuda a los pequeños agricultores con la distribución de sus productos a la compra y venta a nivel nacional del café, como pequeños agricultores que son tratamos de apoyarlos con la distribución de funguicidas, insecticidas, para obtener una mejor producción del café.

2. ¿Puede explicar por que es tan importante la producción y el cuidado del café en el distrito de Chirinos?

Para el distrito de chirinos es muy importante la siembra del café, por que gracias a ellos pertenecemos a uno de los departamentos a nivel nacional de mayor producción de café como es Cajamarca, la mayoría de las familias del distrito de chirinos por no decir todo el distrito se dedica a la siembra y cosecha de este producto como es el café, nosotros como empresa tratamos de apoyarlo a los pequeños agricultores distribuyéndolo su producto a nivel nacional con la compra y venta de café.

3. ¿Cómo gerente general de la empresa, su ingeniero agrónomo como apoya a los agricultores a detectar las diferentes enfermedades en las plantas de café?

El ingeniero agrónomo de la empresa, tiende a prestar sus servicios con los agricultores en la elaboración de insecticidas y recomendándoles abonos eficaces contra las enfermedades para las plantas de café, hay días que hacemos visitas a los diferentes agricultores para ver cómo van con la producción de café, tratamos de apoyarlo eficazmente con la distribución de insecticidas y fungicidas.

4. ¿En los cultivos de café que enfermedades se conoce?

Existe una variedad de enfermedades que atacan a las hojas de café, depende mayormente del cuidado de las plantas y usar abonos adecuados, entre las enfermedades más frecuentes en el distrito de Chirinos tenemos a la roya de la hoja, phoma de la hoja, mancha de hierro, etc.

5. Por su experiencia en el cultivo de café, ¿Qué utilizan los agricultores de café para detectar las enfermedades en las plantas de café, sería de mucha utilidad el uso de una aplicación móvil?

Bueno algunos agricultores lo reconocen por la percepción visual porque tienen experiencia en la siembra del café, pero hay agricultores que tienden a confundir las enfermedades de café, administrándolo diferentes tipos de agroquímicos y matando a la planta, es por ello que sería de mucha utilidad la ayuda de tecnología hoy en día para poder minimizar tiempos y ahorro en gastos de detección de enfermedades, es decir que nos daría gusto poder ayudar a contribuir con la elaboración de la aplicación móvil para ayudar a los pequeños agricultores.

EXPORTADORA DE CAFÉ
"CASMAR" E.I.R.L.

Luis Castillo Martínez
GERENTE

LUIS CASTILLO MARTÍNEZ
Gerente General

ANEXO-03: Encuesta realizada a los agricultores de la provincia de chirinos.

Encuesta Realizada a los agricultores del distrito de chirinos

1. ¿Usted como agricultor hace cuanto tiempo es que cosecha el café?

- 1 - 3 años
- 3 - 5 años
- 5 - 10 años
- 10 - a más años

2. ¿Usted cuenta con un dispositivo móvil con sistema operativo Android?

- Si
- No

3. ¿Se encuentra satisfecho con la producción de café en sus parcelas?

- Si
- No

4. ¿Tiene conocimiento sobre los tipos de enfermedades que atacan al café?

- Si
- No

5. ¿De qué forma usted identifica las diferentes enfermedades en el cultivo del café?

- Observación y Experiencia
- Método Tecnológico
- Recurre a un experto
- otro: _____

6. ¿Cuánto tiempo demora usted en identificar una enfermedad en el cultivo de café?

- 15 minutos
- 30 minutos
- 45 minutos
- 60 minutos
- otro: _____

7. ¿Cuál es el costo promedio que usted gasta para identificar una enfermedad en las plantas de café?

- 30 soles
- 50 soles
- 100 soles
- otro: _____

8. ¿Tiene conocimiento sobre los tipos de agroquímicos que se utilizan para cuidar a las plantas de café?

- Si
- No

9. ¿Tiene en cuenta las dosis de los agroquímicos necesario para usar en las plantas de café?

- Si
- No

10. ¿La aplicación móvil sería útil para usted en cuanto a la identificación de las enfermedades en las plantas de café?

- Si
- No
- Tal vez

Resultados de la encuesta realizada a 80 agricultores en el distrito de chirinos.

Figura 43

Respuesta 1 del Anexo-03



Figura 44

Respuesta 2 del Anexo-03

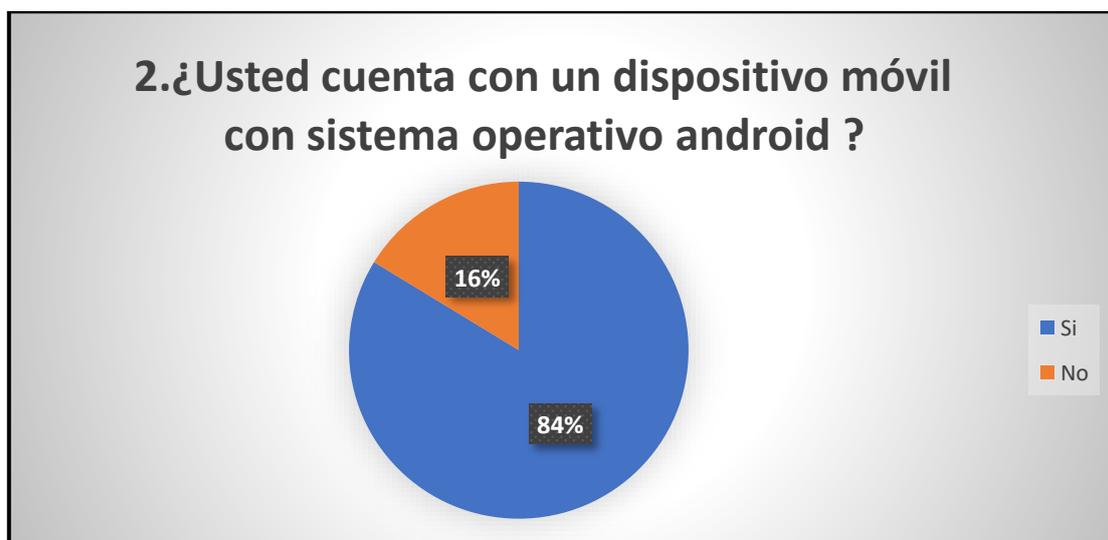


Figura 45

Respuesta 3 del Anexo-03



Figura 46

Respuesta 4 del Anexo-03

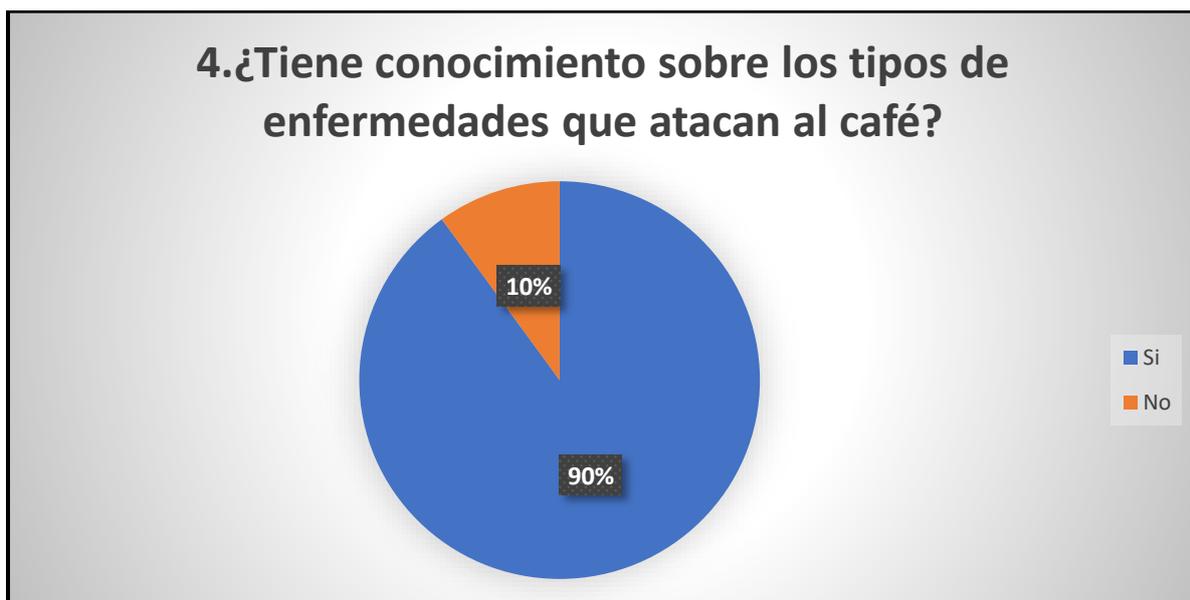


Figura 47

Respuesta 5 del Anexo-03

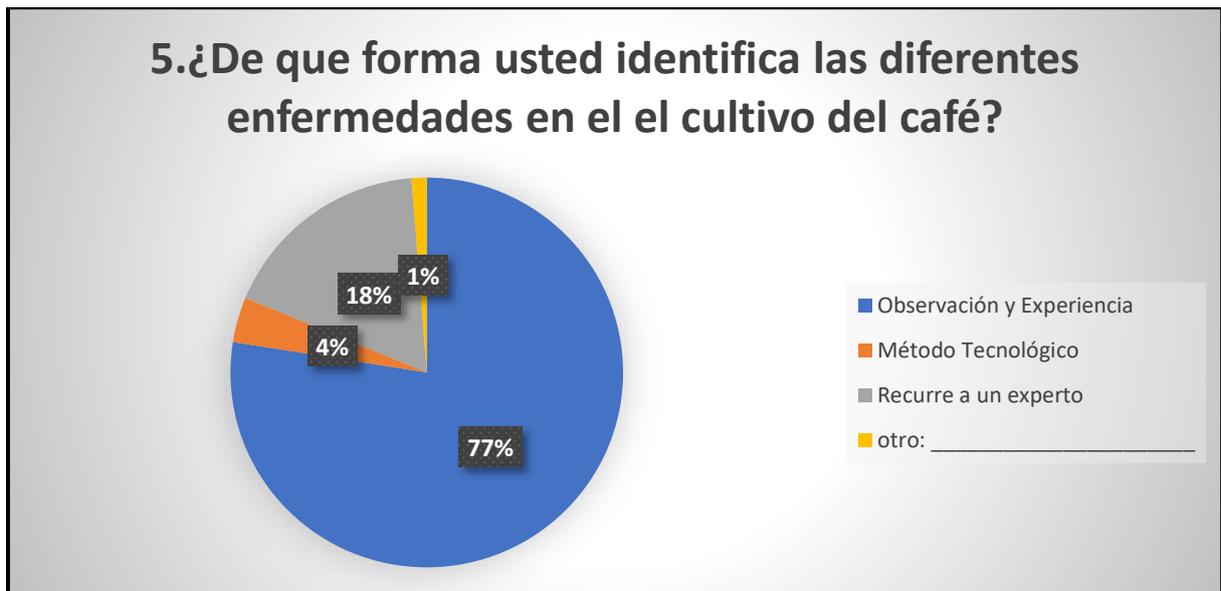


Figura 48

Respuesta 6 del Anexo-03



Figura 49

Respuesta 7 del Anexo-03



Figura 50

Respuesta 8 del Anexo-03

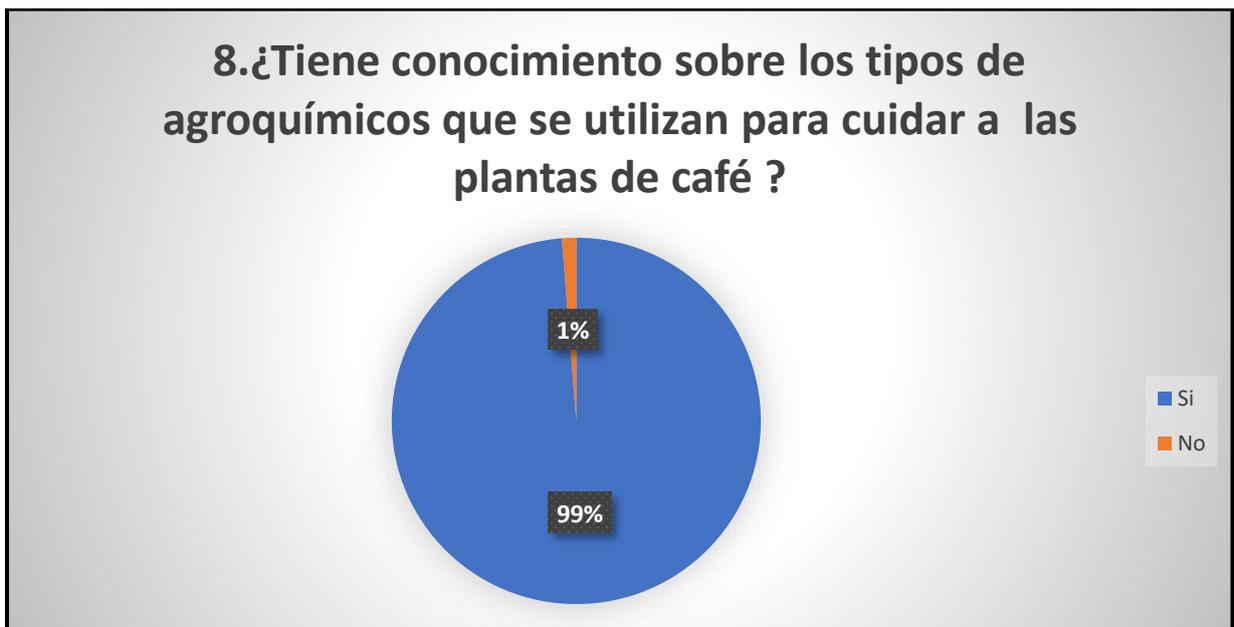


Figura 51

Respuesta 9 del Anexo-03

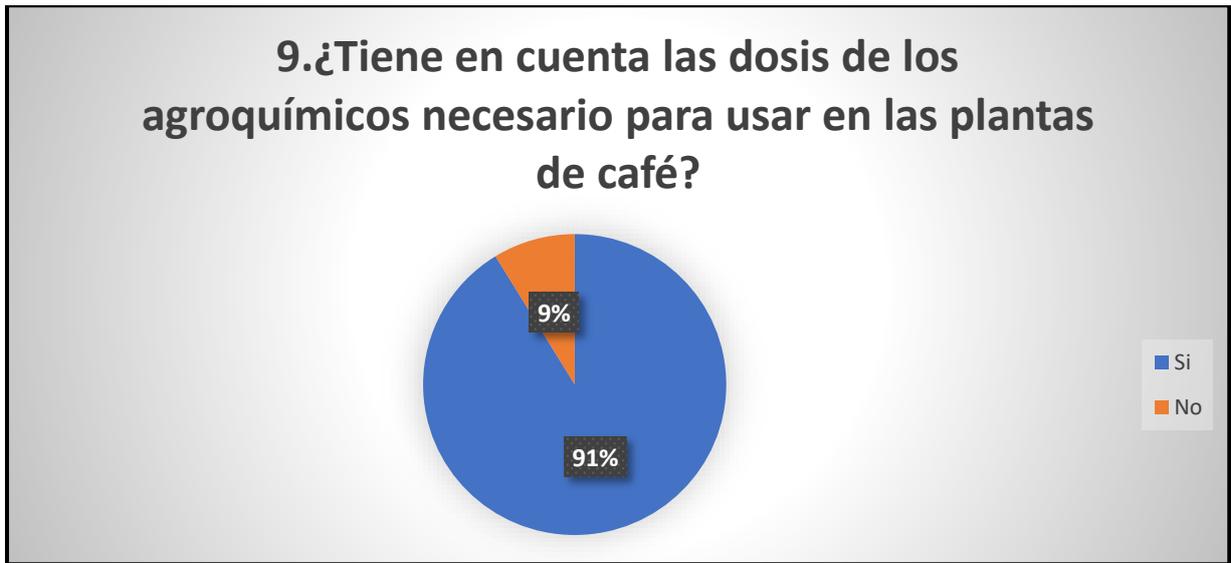
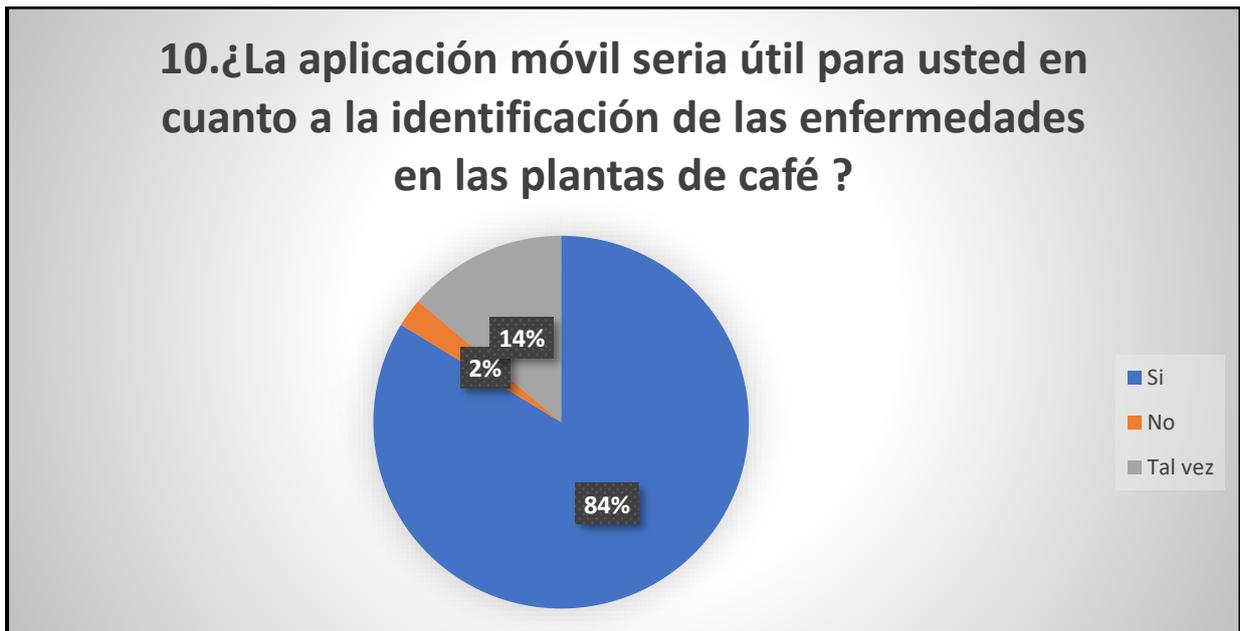


Figura 52

Respuesta 10 del Anexo-03



ANEXO-04: SOLICITUD DIRIGIDA A EXPERTO EN AGRONOMÍA.

SOLICITUD DIRIGIDA A EXPERTO EN:

AGRONOMIA

Ing. Agrónomo.

Presente

Reciba un cordial saludo,

Por medio de la presente, quien suscribe FLORES MENDOZA, Juan Carlos y MEJIA CARHUAJULCA, Gianfranco, identificados con código universitario N° 020155635K y N° 020155640D respectivamente, Bachilleres de la Escuela Profesional de Ingeniería de Sistemas de la Universidad Nacional Pedro Ruiz Gallo, informados de su reconocida formación y experiencia en agronomía, Nos complace dirigimos a usted para solicitarle su valiosa colaboración para la revisión y la validación como EXPERTO de la propuesta para la presente tesis que tiene como título: “*Aplicación Móvil usando redes neuronales convolucionales para la detección de enfermedades en las plantas de café en el distrito de Chirinos, Provincia de San Ignacio, Departamento de Cajamarca*”, para optar el Título Profesional en Ingeniería de Sistemas, agradecería si fuese positiva su aceptación, sírvase a llenar los datos solicitados del recuadro que se encuentra al final del documento.

Agradecemos su aceptación y colaboración para la presente investigación.

Atentamente,

Lambayeque, 10 de noviembre del 2022

Juan Carlos Flores Mendoza

Gianfranco Mejia Carhuajulca

FORMATO DE ACEPTACIÓN DEL EXPERTO

NOMBRE:	ZESAR JHUNIOR URRUTIA ALCANTARA
DNI:	47376057
FIRMA:	 ZESAR JHUNIOR URRUTIA ALCANTARA INGENIERO AGRÓNOMO REG CIP 265534

ANEXO-05: HOJA DE VIDA DEL INGENIERO AGRÓNOMO ZESAR JUNIOR URRUTIA ALCANTARA.



✉ [linkedin.com/in/zesar-urrutia-alc](https://www.linkedin.com/in/zesar-urrutia-alc)

📍 CHICLAYO, PERÚ

• SOBRE MI

Soy una persona proactiva, excelente compañero de equipo, organizado y responsable. Disfruto mucho aprender cosas nuevas, por lo que mi objetivo es un puesto desafiante y dinámico donde poder compartir mi experiencia y a su vez sumar nuevos conocimientos.

• EDUCACIÓN

Universidad Nacional Pedro Ruiz Gallo
AGRÓNOMO

• CURSOS / HABILIDADES

Curso de Informática
ZEGEL IPAE, 2019.

DIPLOMATURA AGRONEGOCIOS
ZEGEL IPAE, 2019-2021

• IDIOMAS

Idioma Inglés
Nivel Escrito Avanzado.

• EXPERIENCIA LABORAL

JEFE DE FUNDO

INVERSIONES YARABAMBA SAC, 2022

- Supervisión y coordinaciones de preparación de campo e instalación de riego.
- Programación, verificación y control de riego y fertilización en plantaciones en desarrollo vegetativo.

JEFE DE SANIDAD

HUERTO GENESIS SAC, 2018-2022

- Ejecución del manejo de fitosanitario en base a evaluaciones de plagas, enfermedades y fenología del cultivo.
- Controlar y analizar los registros y resultados de las evaluaciones fitosanitarias.

JEFE DE CULTIVO

HUERTO GENESIS SAC, 2016-2018

- Programa preventivo para la campaña de pimiento.
- Informes referentes a la programación de actividades del sector sanidad vegetal.

EMPLEADO DE PRÁCTICAS

FUNDO SAN AGUSTIN, 2016

- Manejo integrado de plagas en el cultivo de mango.
- Evaluación de trampas etológicas instaladas en el predio.

ANEXO-06: CODIFICACIÓN DEL MODELO REDES NEURONALES CONVOLUCIONALES

```
#Importacion de librerias

import tensorflow as tf
from keras.preprocessing.image import ImageDataGenerator
from keras.preprocessing import image
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout, BatchNormalization,
Activation
from tensorflow.keras.optimizers import Adam
from keras.callbacks import EarlyStopping, ModelCheckpoint
import numpy as np
import keras.backend as Keras
from tensorflow import keras
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.metrics import confusion_matrix
import seaborn as sn; sn.set(font_scale=1)

Keras.clear_session()

# Creamos los generadores de imagenes de entrenamiento, validación y prueba
# Reescalamos los pixeles con valores entre 0 y 1
# aplicamos transformaciones a la imagenes

train_datagen = ImageDataGenerator(
    rotation_range=90, #rotar la imagenes 90°
    rescale= 1./ 255,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range= 0.2, #transformaciones de corte
    zoom_range= 0.2, #zoom aleatorio dentro de las imagenes
    vertical_flip= True, #Voltear la imagen de forma vertical
    horizontal_flip= True, #Voltear la imagen de forma Horizontal
    dtype = np.uint8 #tipo de dato uint8
)

datagen = ImageDataGenerator(rescale= 1./ 255)

train_generator = train_datagen.flow_from_directory(
    directory= './input/dataset-completo-31-05-2022/tesis_datashet_android_nuevo/train',
    target_size= (128, 128), # tamaño de las imagenes
    color_mode= "rgb", # 3 canales
    batch_size= 32,
    class_mode= "categorical" # one-hot encoded
)

valid_generator = datagen.flow_from_directory(
    directory= './input/dataset-completo-31-05-2022/tesis_datashet_android_nuevo/val',
    target_size= (128, 128), # tamaño de las imagenes
    color_mode= "rgb", # 3 canales
    batch_size= 32,
    class_mode= "categorical"# one-hot encoded
)

test_generator = datagen.flow_from_directory(
    directory= './input/dataset-completo-31-05-2022/tesis_datashet_android_nuevo/test',
    target_size= (128, 128), # tamaño de las imagenes
    color_mode= "rgb", # 3 canales
    batch_size= 30,
    class_mode= "categorical",# one-hot encoded
    shuffle= False
)

# Parámetros
STEP_SIZE_TRAIN= train_generator.n // train_generator.batch_size # Número de pasos que ejecutaremos
para entrenar
STEP_SIZE_VALID= valid_generator.n // valid_generator.batch_size # Número de pasos que ejecutaremos
para validaciones
STEP_SIZE_TEST= test_generator.n // test_generator.batch_size # Número de pasos que ejecutaremos para
pruebas
NUM_CLASSES = train_generator.num_classes # Número de clases totales para la clasificación

#Data Augmentation
from keras.preprocessing.image import ImageDataGenerator, array_to_img, img_to_array, load_img
img = load_img('/kaggle/input/dataset-completo-31-05-2022/tesis_datashet_android_nuevo/train/Phoma de
Hoja/22 (10).jpg') #cargar imagen muestra
x = img_to_array(img) #array
x = x.reshape((1,) + x.shape)
i = 0
for batch in train_datagen.flow(x, batch_size=1, save_to_dir='/kaggle/working', save_prefix='datagen7',
save_format='jpg'): #directorio
    i += 1
    if i > 5:
        break
```

```

#Optimizador

EPOCHS = 20
optimizers_list = [
    tf.keras.optimizers.Adadelta(),
    tf.keras.optimizers.Adam(),
    tf.keras.optimizers.Ftrl(),
    tf.keras.optimizers.Nadam(),
    tf.keras.optimizers.SGD()
]

results = {}
history = {}
for optimizer in optimizers_list:

    model = tf.keras.Sequential()

    # BLOQUE 1
    model.add(Conv2D(32, (3, 3), input_shape= (128, 128, 3), activation= 'relu')) # 32 filtros con un
    tamaño de 3x3 y función de activación ReLu
    model.add(Conv2D(32, (3, 3), activation= 'relu')) # 32 filtros con un tamaño de 3x3 y función de
    activación ReLu
    model.add(MaxPooling2D((2, 2))) # Tamaño de la ventana de agrupación -> 2x2

    # BLOQUE 2
    model.add(Conv2D(64, (3, 3), activation= 'relu')) # 64 filtros con un tamaño de 3x3 y función de
    activación ReLu
    model.add(Conv2D(64, (3, 3), activation= 'relu')) # 64 filtros con un tamaño de 3x3 y función de
    activación ReLu
    model.add(MaxPooling2D((2, 2))) # Tamaño de la ventana de agrupación -> 2x2

    # BLOQUE 3
    model.add(Conv2D(128, (3, 3), activation= 'relu')) # 128 filtros con un tamaño de 3x3 y función de
    activación ReLu
    model.add(MaxPooling2D((2, 2))) # Tamaño de la ventana de agrupación -> 2x2

    # BLOQUE 4
    model.add(Conv2D(256, (3, 3), activation= 'relu')) # 256 filtros con un tamaño de 3x3 y función de
    activación ReLu
    model.add(MaxPooling2D((2, 2))) # Tamaño de la ventana de agrupación -> 2x2

    # BLOQUE 5
    model.add(Conv2D(512, (3, 3), activation= 'relu')) # 512 filtros con un tamaño de 3x3 y función de
    activación ReLu
    model.add(MaxPooling2D((2, 2))) # Tamaño de la ventana de agrupación -> 2x2

    # BLOQUE 6
    model.add(Flatten()) # Convertimos los elementos de la matriz de imágenes en un array de una sola
    dimensión

    #BLOQUE 7
    model.add(Dense(1024, activation= 'relu')) # Fully connected
    model.add(Dropout(0.4)) # Desactivamos un 40% de las neuronas para reducir el sobreajuste
    model.add(Dense(512, activation= 'relu')) # Fully connected
    model.add(Dropout(0.5)) # Desactivamos un 50% de las neuronas para reducir el sobreajuste
    model.add(Dense(NUM_CLASSES, activation= 'relu')) # Fully connected -> número de neuronas = número
    de clases

    model.compile(loss='categorical_crossentropy',
                  optimizer=optimizer,
                  metrics=['accuracy'])
    optimizer_key = str(type(optimizer).__name__)
    history[optimizer_key] = model.fit(train_generator,
                                      steps_per_epoch= STEP_SIZE_TRAIN,
                                      validation_data= valid_generator,
                                      validation_steps= STEP_SIZE_VALID,
                                      epochs= EPOCHS)

    results[optimizer_key] = {}
    results[optimizer_key]["loss"] = history[optimizer_key].history['loss'][EPOCHS - 1]
    results[optimizer_key]["val_loss"] = history[optimizer_key].history['val_loss'][EPOCHS - 1]

df = pd.DataFrame(results)
df.plot.bar()

plt.figure()
plt.xlabel('epoch')
plt.ylabel('Loss')
for optimizer in history:
    hist = pd.DataFrame(history[optimizer].history)
    plt.plot(history[optimizer].epoch, np.array(hist['loss']),
             label='Train Loss '+optimizer)
    plt.plot(history[optimizer].epoch, np.array(hist['val_loss']),
             label = 'Val loss '+optimizer)
plt.legend()

```

```

# Analizamos el porcentaje de imágenes de cada categoría
train_images_per_class = np.unique(train_generator.classes, return_counts= True)

plt.pie(train_images_per_class[1],
        labels= train_generator.class_indices.keys(),
        autopct= '%1.1f%%')

plt.title('Proporción de imágenes por categoría \n')
plt.show()

# modelo
model= tf.keras.Sequential()

# BLOQUE 1
model.add(Conv2D(32, (3, 3), input_shape= (128, 128, 3), activation= 'relu')) # 32 filtros con un
tamaño de 3x3 y función de activación ReLU
model.add(Conv2D(32, (3, 3), activation= 'relu')) # 32 filtros con un tamaño de 3x3 y función de
activación ReLU
model.add(MaxPooling2D((2, 2))) # Tamaño de la ventana de agrupación -> 2x2

# BLOQUE 2
model.add(Conv2D(64, (3, 3), activation= 'relu')) # 64 filtros con un tamaño de 3x3 y función de
activación ReLU
model.add(Conv2D(64, (3, 3), activation= 'relu')) # 64 filtros con un tamaño de 3x3 y función de
activación ReLU
model.add(MaxPooling2D((2, 2))) # Tamaño de la ventana de agrupación -> 2x2

# BLOQUE 3
model.add(Conv2D(128, (3, 3), activation= 'relu')) # 128 filtros con un tamaño de 3x3 y función de
activación ReLU
model.add(MaxPooling2D((2, 2))) # Tamaño de la ventana de agrupación -> 2x2

# BLOQUE 4
model.add(Conv2D(256, (3, 3), activation= 'relu')) # 256 filtros con un tamaño de 3x3 y función de
activación ReLU
model.add(MaxPooling2D((2, 2))) # Tamaño de la ventana de agrupación -> 2x2

# BLOQUE 5
model.add(Conv2D(512, (3, 3), activation= 'relu')) # 512 filtros con un tamaño de 3x3 y función de
activación ReLU
model.add(MaxPooling2D((2, 2))) # Tamaño de la ventana de agrupación -> 2x2

# BLOQUE 6
model.add(Flatten()) # Convertimos los elementos de la matriz de imágenes en un array de una sola
dimensión

# BLOQUE 7
model.add(Dense(1024, activation= 'relu')) # Fully connected
model.add(Dropout(0.4)) # Desactivamos un 40% de las neuronas para reducir el sobreajuste
model.add(Dense(512, activation= 'relu')) # Fully connected
model.add(Dropout(0.5)) # Desactivamos un 50% de las neuronas para reducir el sobreajuste
model.add(Dense(NUM_CLASSES, activation= 'relu')) # Fully connected -> número de neuronas = número de
clases

model.compile(optimizer= Adam(lr= 0.0001), #factor de aprendizaje
              loss= 'categorical_crossentropy',
              metrics= 'accuracy')

model.summary()

plot_model(model, show_shapes=True, to_file= '/kaggle/working/imgane2.jpg', show_layer_names=True)

import visualkeras
from PIL import ImageFont
visualkeras.Layered_view(model, legend=True)

Model = model.fit(train_generator,
                  steps_per_epoch= STEP_SIZE_TRAIN,
                  validation_data= valid_generator,
                  validation_steps= STEP_SIZE_VALID,
                  epochs= 20,
                  callbacks = [EarlyStopping(monitor= 'val_loss', patience= 3),
                              ModelCheckpoint (filepath = '/kaggle/working/Modelo.h5', monitor = 'val_loss',
                              save_best_only = True)] # Monorizamos la función de pérdida de los datos de validación
                  )

# Creamos dos gráficas y comparamos la exactitud y la pérdida con respecto a los datos de entrenamiento
y validación

# Gráfica accuracy
plt.plot(Model.history['accuracy'])
plt.plot(Model.history['val_accuracy'])
plt.title('Model Accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train', 'Validation'], loc='lower right')
plt.show()

# Gráfica loss
plt.plot(Model.history['loss'])
plt.plot(Model.history['val_loss'])
plt.title('Model Loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Validation'], loc='lower left')
plt.show()

```

```

# Guardamos el modelo entrenado

model.save('/kaggle/working/ModeloI-16-12-2022.h5')

# Convertimos el modelo y exportamos el modelo a TensorFlow
import tensorflow as tf
model = tf.keras.models.load_model('/kaggle/working/ModeloII-16-12-2022.h5')
converter = tf.lite.TFLiteConverter.from_keras_model(model)
tflmodel = converter.convert()
file = open( '/kaggle/working/ModeloII2-16-12-2022.tflite' , 'wb' )
file.write( tflmodel )

# Cargamos el modelo

modelLoad = keras.models.load_model('/kaggle/working/ModeloII-16-12-2022.h5')

# Evaluamos el modelo con los datos de prueba

results = modelLoad.evaluate(test_generator, steps= STEP_SIZE_TEST)
print("Loss: {} Accuracy: {}".format(results[0], results[1] * 100))

# Analizamos las predicciones de los datos de test

predictions = modelLoad.predict(test_generator) # Vector de probabilidades
pred_labels = np.argmax(predictions, axis = 1) # Obtenemos la predicción con mas probabilidad
class_names = test_generator.class_indices.keys() # Nombre de las etiquetas

CM = confusion_matrix(test_generator.classes, pred_labels) # Matriz de confusión

ax = plt.axes()
sn.heatmap(CM, annot= True,
            annot_kws={"size": 10},
            xticklabels= class_names,
            yticklabels= class_names,
            ax = ax,
            fmt= "d")
ax.set_title('Matriz de confusión')
plt.show()

# Función para obtener la etiqueta

def getcode(n) :
    for x , y in train_generator.class_indices.items() :
        if n == y :
            return x

# Usamos el modelo para predecir una imagen en concreto

img = image.load_img('/kaggle/input/dataset-completo-31-05-2022/tesis_datashet_android_nuevo/test/Roya
de Hoja/21 (7950).jpg', target_size = (128, 128))
img = image.img_to_array(img)
img = np.expand_dims(img, axis = 0)

predict = modelLoad.predict(img)
print(predict)
print('Predicción de la imagen: ' + getcode(np.argmax(predict)))

```



**ACTA DE SUSTENTACIÓN VIRTUAL
 N° 005-2023-FICSA-D**



Siendo las 8:00 am horas del día 29 de mayo del 2023, se reunieron vía plataforma virtual, <https://meet.google.com/bnj-ysgd-hxt>, los miembros de jurado de la tesis titulada: **APLICACIÓN MÓVIL USANDO REDES NEURONALES CONVOLUCIONALES PARA LA DETECCIÓN DE ENFERMEDADES EN LAS PLANTAS DE CAFÉ EN EL DISTRITO DE CHIRINOS, PROVINCIA DE SAN IGNACIO, DEPARTAMENTO DE CAJAMARCA** con código de proyecto IS_V_2022_006, designado por **Resolución Decanal Virtual N° 154-2022-UNPRG-FICSA** con la finalidad de Evaluar y Calificar la sustentación de la tesis antes mencionada, conformado por los siguientes docentes:

DR. ING. EDWAR RONALD HARO MALDONADO	PRESIDENTE
MG. ING. ROBERT EDGAR PUICAN GUTIERREZ	SECRETARIO
DR. ING. JUAN ELIAS VILLEGAS CUBAS	VOCAL

Asesorado por **MG. ING. ROBERTO CARLOS ARTEAGA LORA**

El acto de sustentación fue autorizado por **OFICIO VIRTUAL N° 096-2023-UIFICSA**, la tesis fue presentada y sustentada por los Bachilleres: **FLORES MENDOZA JUAN CARLOS** y **MEJIA CARHUAJULCA, GIANFRANCO**, tuvo una duración de 30 minutos. Después de la sustentación, y absueltas las preguntas y observaciones de los miembros del jurado; se procedió a la calificación respectiva.

FLORES MENDOZA JUAN CARLOS	18	DIECIOCHO	MUY BUENO
MEJIA CARHUAJULCA, GIANFRANCO	18	DIECIOCHO	MUY BUENO

Por lo que quedan **APTOS** para obtener el Título Profesional de **INGENIERO DE SISTEMAS** de acuerdo con la Ley Universitaria 30220 y la normatividad vigente de la Facultad de Ingeniería Civil De Sistemas y de Arquitectura de la Universidad Nacional Pedro Ruiz Gallo.

Siendo las 9.30 a.m. Se dio por concluido el presente acto académico, dándose conformidad al presente acto, con la firma de los miembros del jurado.

DR. ING. EDWAR RONALD HARO MALDONADO
PRESIDENTE

MG. ING. ROBERT EDGAR PUICAN GUTIERREZ
SECRETARIO

DR. ING. JUAN ELIAS VILLEGAS CUBAS
VOCAL

MG. ING. ROBERTO CARLOS ARTEAGA LORA
ASESOR



DR. ING. SERGIO BRAVO IDROGO
DECANO



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO
FACULTAD DE INGENIERÍA CIVIL, DE SISTEMAS Y ARQUITECTURA
UNIDAD DE INVESTIGACIÓN



“Año de la unidad, la paz y el desarrollo”

CONSTANCIA DE APROBACIÓN DE ORIGINALIDAD DE TESIS

Según Res. N° 659-2020-R

Yo, ARTEAGA LORA ROBERTO CARLOS, asesor de tesis de los bachilleres:

FLORES MENDOZA, Juan Carlos
MEJÍA CARHUAJULCA, Gianfranco

TITULADA:

TESIS: “Aplicación Móvil usando redes neuronales convolucionales para la detección de enfermedades en las plantas de café en el distrito de Chirinos, Provincia de San Ignacio, Departamento de Cajamarca”

Luego de la revisión exhaustiva del documento constato, que el mismo tiene un índice de similitud de **12%** verificable en el reporte de similitud del programa TURNITIN.

El suscrito analizó dicho reporte y concluyó que, cada una de las coincidencias detectadas **NO CONSTITUYEN PLAGIO**. A mi leal saber y entender, la tesis cumple con todas las normas para el uso de citas y referencias establecidas por la Universidad Nacional Pedro Ruiz Gallo.

Se expide la presente según lo dispuesto en la Resolución N° 659-2020-R, de fecha 8 de setiembre de 2020, que aprueba la directiva para la evaluación de originalidad de los documentos académicos, de investigación formativa y para la obtención de Grados y Títulos de la UNPRG:

Lambayeque, 31 de enero de 2023

Atentamente,

MA. ING. ROBERTO CARLOS ARTEAGA LORA
DNI 16755764

Se adjunta:
Recibo digital de Turnitin
Revisión de informe en Turnitin



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO
FACULTAD DE INGENIERÍA CIVIL, DE SISTEMAS Y ARQUITECTURA
UNIDAD DE INVESTIGACIÓN



Recibo digital

Este recibo confirma que su trabajo ha sido recibido por Turnitin. A continuación podrá ver la información del recibo con respecto a su entrega.

La primera página de tus entregas se muestra abajo.

Autor de la entrega: Juan Flores Mendoza
Título del ejercicio: Informe borrador de tesis
Título de la entrega: Aplicación Móvil usando redes neuronales convolucionales ...
Nombre del archivo: PROYECTO_FINAL_FLORES_MEJIA.pdf
Tamaño del archivo: 4,84M
Total páginas: 101
Total de palabras: 12,229
Total de caracteres: 75,308
Fecha de entrega: 31-ene.-2023 05:10p. m. (UTC-0500)
Identificador de la entre... 2003618651

 UNIVERSIDAD NACIONAL
PEDRO RUIZ GALLO 

FACULTAD DE INGENIERÍA CIVIL, SISTEMAS Y
ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE
SISTEMAS

TESIS

"Aplicación Móvil usando redes neuronales convolucionales
para la detección de enfermedades en las plantas de café en el
distrito de Chirinos, Provincia de San Ignacio, Departamento
de Cajamarca".

Para obtener el título profesional de:
INGENIERO DE SISTEMAS

AUTORES:
FLORES MENDOZA, Juan Carlos
MEHA CARHUARILCA, Gianfranco

ASESOR:
MG. ING. ARTEAGA LORA, Roberto Carlos

LAMBAYEQUE-PERÚ
2023


Ing. Roberto Arteaga



Aplicación Móvil usando redes neuronales convolucionales para la detección de enfermedades en las plantas de café en el distrito de Chirinos, Provincia de San Ignacio, Departamento de Cajamarca

INFORME DE ORIGINALIDAD

12%	10%	1%	6%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	hdl.handle.net Fuente de Internet	2%
2	repositorio.unprg.edu.pe Fuente de Internet	1%
3	Submitted to Universidad Industrial de Santander UIS Trabajo del estudiante	1%
4	repositorio.espam.edu.ec Fuente de Internet	1%
5	Submitted to Universidad Nacional de San Cristóbal de Huamanga Trabajo del estudiante	1%
6	repositorio.untrm.edu.pe Fuente de Internet	1%
7	repositorio.utn.edu.ec Fuente de Internet	<1%

Ing. Roberto Arteaga