



UNIVERSIDAD TÉCNICA DE COTOPAXI
FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS
CARRERA DE INGENIERÍA EN INFORMÁTICA Y SISTEMAS
COMPUTACIONALES

PROYECTO DE INVESTIGACIÓN

TEMA:

**“DESARROLLO DE UN PROTOTIPO PARA EL RECONOCIMIENTO
DE PLAGAS EN PLANTACIONES DE BABACO BAJO
INVERNADERO UTILIZANDO REDES NEURONALES”**

Proyecto de investigación presentado previo a la obtención del Título de
Ingenieros en Informática y Sistemas Computacionales

AUTORES:

Gamboa Murillo Patricio Alejandro

López Miniguano Isaac Samuel

DIRECTOR DE TESIS:

Ing. Mg. William Manuel Villa Quishpe

LATACUNGA – ECUADOR

2021

DECLARACIÓN DE AUTORÍA

Nosotros, **Gamboa Murillo Patricio Alejandro** con C.I.: **180483386-9** y **López Miniguano Isaac Samuel** con C.I.: **185059081-9**, ser los autores del presente proyecto de Investigación: **“DESARROLLO DE UN PROTOTIPO PARA EL RECONOCIMIENTO DE PLAGAS EN PLANTACIONES DE BABACO BAJO INVERNADERO UTILIZANDO REDES NEURONALES”**, siendo el Ing. MSc. **William Manuel Villa Quishpe**, tutor del presente trabajo, eximo expresamente a la Universidad Técnica de Cotopaxi y a sus representantes legales de posibles reclamos o acciones legales.

Además, certificamos que las ideas, conceptos, procedimientos y resultados vertidos en el presente trabajo investigativo, son de nuestra exclusiva responsabilidad.

Atentamente,

.....
Patricio Alejandro Gamboa Murillo
CI: 180483386-9

.....
Isaac Samuel López Miniguano
CI: 185059081-9

AVAL DEL TUTOR DE PROYECTO DE TITULACIÓN

En calidad de Tutor del Trabajo de Investigación con el título:

“DESARROLLO DE UN PROTOTIPO PARA EL RECONOCIMIENTO DE PLAGAS EN PLANTACIONES DE BABACO BAJO INVERNADERO UTILIZANDO REDES NEURONALES”, de los estudiantes: **GAMBOA MURILLO PATRICIO ALEJANDRO** y **LÓPEZ MINIGUANO ISAAC SAMUEL** de la Carrera de Ingeniería en Informática y Sistemas Computacionales, considero que dicho Informe Investigativo cumple con los requerimientos metodológicos y aportes científico-técnicos suficientes para ser sometidos a la evaluación del Tribunal de Validación de Proyecto que el Honorable Consejo Académico de la Facultad de Ciencias de la Ingeniería y Aplicadas de la Universidad Técnica de Cotopaxi designe, para su correspondiente estudio y calificación.

Latacunga, Agosto del 2021

.....
Ing. MSc. William Manuel Villa Quishpe

Tutor

C.C.: 180338695-0

APROBACIÓN DEL TRIBUNAL DE TITULACIÓN

En calidad de Tribunal de Lectores, aprueban el presente Informe de Investigación de acuerdo a las disposiciones reglamentarias emitidas por la Universidad Técnica de Cotopaxi, y por la Facultad de **CIENCIAS DE LA INGENIERÍA Y APLICADAS**; por cuanto, los postulantes: **Gamboa Murillo Patricio Alejandro** con cedula de ciudadanía C.I: **180483386-9** y **López Miniguano Isaac Samuel** con cédula de ciudadanía C.I: **185059081-9**, con el título del proyecto de investigación: **“DESARROLLO DE UN PROTOTIPO PARA EL RECONOCIMIENTO DE PLAGAS EN PLANTACIONES DE BABACO BAJO INVERNADERO UTILIZANDO REDES NEURONALES”**, ha considerado las recomendaciones emitidas oportunamente y reúne los méritos suficientes para ser sometido al acto de Sustentación del Proyecto.

Por lo antes expuesto, se autoriza realizar los empastados correspondientes, según la normativa institucional

Latacunga, Agosto del 2021

.....
Ing. Msc. Mayra Albán
C.C: 0502311988

.....
Ing. Msc. Juan Chancusig
C.C: 0502275779

.....
Ing. Msc. Alex Llano
C.C: 0502589864

AVAL DE ACEPTACIÓN

Mediante el presente pongo a consideración que los señores estudiantes **Gamboa Murillo Patricio Alejandro** y **López Miniguano Isaac Samuel**, realizaron su tesis a beneficio de la UNIVERSIDAD TÉCNICA DE COTOPAXI con el tema: **“DESARROLLO DE UN PROTOTIPO PARA EL RECONOCIMIENTO DE PLAGAS EN PLANTACIONES DE BABACO BAJO INVERNADERO UTILIZANDO REDES NEURONALES”**, trabajo que fue presentado y probado de manera satisfactoria.

.....

Ing. Msc. William Manuel Villa Quishpe

C.C: 180338695-0

AGRADECIMIENTO

En primer lugar, deseo agradecer a Dios por darme la fuerza y las ganas de no desmayar en este arduo trayecto y por darme la oportunidad de cumplir una meta más en mi camino.

A mis padres por ser el pilar fundamental brindándome su apoyo incondicional en todo momento, a mi novia que siempre fue mi motivo de inspiración para salir adelante en esta etapa de preparación. A la Universidad Técnica de Cotopaxi y a la Facultad de Ciencias de la Ingeniería y Aplicadas extendiendo mi más sincero agradecimiento por abrirme las puertas y darme la oportunidad de obtener una profesión, a los docentes que son nuestra guía en especial al Ing. Edwin Quinatoa quien confió en mí y me dio la oportunidad de seguir luchando por alcanzar y culminar con éxito esta etapa de mi vida.

Patricio Gamboa

Mi agradecimiento a quien ha forjado mi camino y me ha dirigido por el sendero correcto, Dios, el que en todo momento está conmigo ayudándome a aprender de mis errores. Mi Dios eres quien guía por siempre el destino de mi vida.

A mis padres por el apoyo y esfuerzo durante el ciclo de estudio, ya que me brindaron apoyo emocional necesario para seguir adelante.

A mi esposa y a mi hija por su apoyo incondicional. Me brindaron su apoyo, me comprendieron, teniendo tolerancia e infinita paciencia y cediendo su tiempo para que “Papá estudie” y permitir así llevar adelante un proyecto que pasó de ser una meta personal a un emprendimiento más de familia. A ellos mi eterno amor y gratitud.

A la Universidad Técnica de Cotopaxi por permitirme demostrar todo mi esfuerzo y sacrificio durante más de 5 años de mi vida, brindándome experiencias que lo llevare siempre presente en mi vida cotidiana.

Isaac López

DEDICATORIA

Dedico este proyecto de tesis a Dios, a mis padres a mis hermanos y a mi novia, quienes me han apoyado durante la formación de mi carrera profesional, y a todos aquellos que me extendieron la mano en la finalización de este trayecto. Expreso estas palabras desde lo más infinito de mi corazón y anhelo con ellas, dar a entender el aprecio, cariño y amor a todos aquellos que me apoyaron, por lo antes expresado quiero dedicarles este proyecto, en el cual se refleja todo lo aprendido en mi etapa universitaria.

Patricio Gamboa

Se la dedico al forjador de mi camino, a mi padre celestial, el que me acompaña y siempre me levanta de mis continuas caídas. A Dios va todas mis metas propuestas en el trayecto de mi vida.

Mis padres por haberme inducido como la persona que soy en la actualidad, me formaron con reglas, me brindaron algunas libertades, pero al final de cuentas, me motivaron constantemente para alcanzar mis anhelos. Muchos de mis logros se los debo a ustedes entre los que se incluye este.

Posiblemente en este momento no entiendas mis palabras, pero para cuando seas capaz, quiero que te des cuenta de lo que significas para mí, eres la razón que me levanta cada día y esforzarme por el presente y el mañana, eres mi principal motivación.

Como en todos mis logros, en este está presente, muchas gracias Eliana Monserrath, hijita de mi corazón.

Tu ayuda ha sido fundamental, has estado conmigo incluso en los momentos más turbelosos, A mi esposa, hubo momentos difíciles en nuestra relación, pero lo superamos juntos, créeme valió la pena. Uno de ellos fue la culminación de esta etapa de estudio, en la cual recibí tu total apoyo en todo el sentido de la palabra. Te amo Lorena.

Isaac López

UNIVERSIDAD TÉCNICA DE COTOPAXI

FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS

TITULO: “DESARROLLO DE UN PROTOTIPO PARA EL RECONOCIMIENTO DE PLAGAS EN PLANTACIONES DE BABACO BAJO INVERNADERO UTILIZANDO REDES NEURONALES”

Autores:

Gamboa Murillo Patricio Alejandro

López Miniguano Isaac Samuel

RESUMEN

El presente proyecto de investigación está enfocado en la detección de plagas que se dan en las plantas de babaco, es decir se hace un análisis técnico por medio de expertos y a través de revisiones bibliográficas considerando para este proyecto el procesamiento de imágenes mediante el entrenamiento con redes neuronales convolucionales cuyos resultados se reflejaran en una interfaz móvil, la misma que fue diseñada para los agricultores de este tipo de plantaciones.

Para el desarrollo del entrenamiento de la red neuronal se empleó la metodología de desarrollo evolutivo, mediante el uso de herramientas de programación se usaron Google Colab como plataforma de desarrollo, Python como lenguaje de programación, la librería Open CV para el procesamiento de imágenes y Android Studio el cual se utilizó para crear la interfaz móvil. Para el entrenamiento y la prueba de la red neuronal se utilizaron 2500 imágenes de plagas en hojas y frutos basadas en fotografías. Para establecer el tipo de plaga se usó el procesamiento de imágenes en un banco de cincuenta fotografías por plaga mediante dos procesos de entrenamiento de la red neuronal, obteniendo como resultados de efectividad 93,75% y 100% de exactitud.

Palabras Claves: Procesamiento, plagas, Google Colab, OpenCV, Keras.

TECHNICAL UNIVERSITY OF COTOPAXI
FACULTY OF ENGINEERING SCIENCES
AND APPLIED

THEME: “Development of a prototype for the recognition of pests in greenhouse babaco plantations using neural networks”

Authors:

Gamboa Murillo Patricio Alejandro

López Miniguano Isaac Samuel

ABSTRACT

This research project is focused on the detection of pests that occur in babaco plants, that is, a technical analysis is made by experts and through bibliographic reviews considering image processing for this project through training with convolutional neural networks whose results are reflected in a mobile interface, the same one that was designed for farmers in this type of plantation.

For the development of the neural network training, the evolutionary development methodology was used, through the use of programming tools, Google Colab was used as a development platform, Python as a programming language, the Open CV library for image processing and Android Studio which was used to create the mobile interface. For training and testing of the neural network, 2500 images of pests in leaves and fruits based on photographs were used. To establish the type of plague, image processing was used in a bank of fifty photographs per plague through two training processes of the neural network, obtaining 93.75% effectiveness and 100% accuracy as results.

Keywords: Processing, pests, Google Colab, OpenCV, Keras.

ÍNDICE GENERAL

DECLARACIÓN DE AUTORÍA.....	ii
AVAL DEL TUTOR DE PROYECTO DE TITULACIÓN.....	iii
APROBACIÓN DEL TRIBUNAL DE TITULACIÓN	iv
AVAL DE ACEPTACIÓN	v
<i>AGRADECIMIENTO</i>	vi
<i>DEDICATORIA</i>	viii
ABSTRACT.....	xi
1. INFORMACIÓN GENERAL	1
2. INTRODUCCIÓN	3
2.1. EL PROBLEMA	4
2.1.1. Situación Problemática.....	4
2.1.2. Formulación del problema.....	6
2.2. OBJETO Y CAMPO DE ACCION	6
2.2.1. Objeto	6
2.2.2. Campo de acción	6
2.3. BENEFICIARIOS.....	6
2.3.1. Beneficiarios Directos	6
2.3.2. Beneficiarios Indirectos.....	6
2.4. JUSTIFICACION	6
2.5. HIPÓTESIS.....	7
2.6. OBJETIVOS	8
2.6.1. Objetivo General	8
2.6.2. Objetivos Específicos	8
2.7. SISTEMA DE TAREAS.....	7
3. FUNDAMENTACIÓN TEÓRICA.....	8
3.1. BABACO EN INVERNADERO	8
3.1.1. Clasificación botánica	8
3.1.2. Morfología.....	9
3.1.3. Propagación	10
3.2. PLAGAS EN EL BABACO	10
3.2.1. Pulgón (Aphis sp)	10
3.2.2. Ácaros (Tetranychus urticae)	10

3.2.3.	Nemátodos (Melodogine incognita)	10
3.3.	REDES NEURONALES	10
3.3.1.	Ventajas que ofrece la red neuronal	11
3.3.2.	Elementos básicos que componen una red neuronal	12
3.3.3.	Aprendizaje, Validación y Codificación	13
3.3.3.1.	Niveles o capas de una red neuronal.....	13
3.3.3.2.	Tipos de neuronas artificiales	14
3.3.3.3.	Técnicas de decisión	14
3.3.3.4.	Mecanismos de aprendizaje	14
3.3.4.	Principales Topologías	16
3.3.5.	Redes neuronales convolucionales	17
3.4.	VISIÓN ARTIFICIAL	18
3.4.1.	Componentes de la visión artificial	18
3.5.	PHYTON.....	19
3.6.	GOOGLE COLAB	20
3.7.	TENSORFLOW	20
3.8.	OPEN CV	20
3.9.	KERAS.....	20
3.10.	REMOVEBG.....	21
3.11.	GOOGLE RESNET 50.....	21
3.12.	ANDROID STUDIO	21
3.13.	APLICACIÓN MOVIL.....	21
4.	MATERIALES Y MÉTODOS	22
4.1.	Experimental	22
4.2.	De Campo.....	22
4.3.	Documental	22
4.4.	Tecnológica.....	22
4.5.	Métodos de Investigación	22
4.5.1.	Método hipotético deductivo.....	22
4.6.	Técnicas de Investigación	23
4.6.1.	Entrevista.....	23
4.6.2.	Observación.....	23
4.7.	Instrumentos de investigación.....	23
4.7.1.	Ficha bibliográfica.....	23

4.7.2.	Fotografía	23
4.8.	Metodologías de desarrollo de software	24
4.8.1.	Modelo de desarrollo evolutivo.....	24
4.9.	Protocolo para la clasificación y procesamiento digital de imágenes.....	24
4.9.1.	Selección y captura de imágenes.....	24
4.9.2.	Preprocesamiento	24
4.9.3.	Procesamiento.....	24
5.	ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS.....	24
5.1.	Herramientas de programación	24
5.2.	Seguimiento de la metodología de Desarrollo.	25
5.3.	Estados de la metodología de desarrollo.....	25
5.3.1.	Especificación.....	25
5.3.2.	Desarrollo	25
5.3.3.	Validación.....	26
5.4.	Definición de Roles de Equipo	26
5.7.	Desarrollo del esquema integral.....	27
5.8.	Desempeño de redes neuronales en proyectos de visión artificial.....	28
5.8.1.	Aspectos comunes entre las investigaciones	29
5.8.2.	Aspectos no comunes entre las investigaciones	33
5.8.3.	Aspectos comunes entre las investigaciones	33
5.8.4.	Modelo de red neuronal adecuado al problema de investigación.....	34
5.8.5.	Lineamientos de la categorización y procesamiento de imágenes	34
5.9.	Entrenamiento de la red neuronal y pruebas de clasificación.....	35
5.9.1.	Importación de paquetes y librerías.....	35
5.9.2.	Cargar datos (imágenes)	35
5.9.3.	Programar y configurar sobre la arquitectura de aprendizaje profundo ResNet50	37
5.9.4.	Configuración y ejecución del entrenamiento	38
5.9.5.	Resultados del Testing o prueba del modelo	44
5.9.5.1.	Resultado del test de los entrenamientos	45
5.9.5.2.	Estadística de predicciones	45
5.9.6.	Historias de usuario	46
5.10.	Resultado de la validación de la metodología de desarrollo evolutivo ...	48
5.10.1.	Plan de validación de la aplicación (ejecución de pruebas).....	51

5.11.	Validación de la hipótesis.....	53
5.11.1.	Juicio de experto	53
6.	CONCLUSIONES Y RECOMENDACIONES	53
6.1.	Conclusiones	53
6.2.	Recomendaciones.....	54
7.	BIBLIOGRAFÍA.....	55
8.	ANEXOS.....	60
8.1.	Anexo: Hoja de vida de investigadores.....	60
8.2.	Anexo: Formulario de Entrevista	62
	62
8.2.1.	Anexo: Resultado de la entrevista	63
8.3.	Anexo: Plagas que se presentan en las plantaciones de babaco.....	64
8.3.1.	Anexo: Cedula del agricultor Héctor Mecías	65
8.4.	Anexo: Instrumento aplicado a expertos.....	66
8.4.1.	Anexo: Instrumento aplicado a experto en redes neuronales	74
8.5.	Anexo: Gastos indirectos	78
8.6.	Anexo: Gastos directos	78
8.7.	Anexo: Fotografías de toma de muestras por parte del grupo investigador.	
	79	

ÍNDICE TABLAS

Tabla 1. Títulos obtenidos del coordinador.....	2
Tabla 2. Hipótesis.....	8
Tabla 3. Sistema de tareas (planificación de actividades).....	7
Tabla 4. Herramientas de desarrollo	25
Tabla 5. Rol de equipo (equipo de trabajo).....	26
Tabla 6. Rol de equipo (equipo de trabajo).....	26
Tabla 7. Rol de equipo (equipo de trabajo).....	26
Tabla 8. Resultado de la investigación modelos de redes neuronales empleados en proyectos de visión artificial.	32
Tabla 9. Tabulación entre resultados de experto y el entrenamiento de la red neuronal.....	42
Tabla 10. Resultado de los dos entrenamientos	45
Tabla 11. Resultado de las predicciones de la estadística	46
Tabla 12. Historia de usuario 1. Selección de la carpeta dentro de Google drive .	46
Tabla 13. Historia de usuario 2. Selección de imágenes de las carpetas almacenadas en el drive	47
Tabla 14. Historia de usuario 3. Predicción del tipo de plaga.....	48
Tabla 15. Ejecución de pruebas de la historia de usuario número 1.	51
Tabla 16. Ejecución de pruebas de la historia de usuario número 2.	52
Tabla 17. Ejecución de pruebas de la historia de usuario número 3.	52

ÍNDICE FIGURAS

Figura 1. Planta de babaco	8
Figura 2. Propagación del babaco	10
Figura 3. Red neuronal artificial	11
Figura 4. Red neuronal totalmente conectada	12
Figura 5. Comparación de una neurona biológica y una artificial.	12
Figura 6. Esquema proceso de decisión	14
Figura 7. Procesos de la visión artificial	18
Figura 8. Diagrama de bloques de un sistema SVA.....	19
Figura 9. Fotografías tomadas por el grupo de investigadores.	23
Figura 10. Diagrama del modelo principal.	27
Figura 11. Tipos de plagas y cantidad de fotografías capturadas y procesadas	34
Figura 12. Importación de librerías, notebook de entrenamiento de la red neuronal	35
Figura 13. Vinculación a Google drive, notebook de entrenamiento de la red neuronal.....	35
Figura 14. Función load_dataset, notebook de entrenamiento de la red neuronal	36
Figura 15. Inicialización de matrices y carga de datos, notebook de entrenamiento de la red neuronal	36
Figura 16. Bloque de identidad, notebook de entrenamiento de la red neuronal	37
Figura 17. Bloque convolucional, notebook de entrenamiento de la red neuronal	38
Figura 18. Proceso ResNet50, notebook de entrenamiento de la red neuronal.....	38
Figura 19. Configuración del proceso de argumentación, notebook de entrenamiento de la red neuronal	39
Figura 20. Configuración del entrenamiento, notebook de entrenamiento de la red neuronal.....	39
Figura 22. Resultado de la evaluación del primer rendimiento del modelo, notebook de entrenamiento de la red neuronal	40
Figura 23. Resultado de la evaluación del segundo rendimiento del modelo, notebook de entrenamiento de la red neuronal	40
Figura 24. Cuadro estadístico de la tabulación	42
Figura 25. Resultado de la efectividad alcanzada del entrenamiento (pulgonas) ..	42
Figura 26. Resultado de la efectividad alcanzada del entrenamiento (pulgonas) ..	43
Figura 27. Predicciones del modelo, notebook de entrenamiento de la red neuronal	43
Figura 28. Modelo almacenado, notebook de entrenamiento de la red neuronal ..	43

Figura 29. Resultado de la exactitud	44
Figura 30. Resultado de la pérdida del modelo	44
Figura 31. Resultado de la exactitud	45
Figura 32. Resultado de la pérdida del modelo	45
Figura 33. Ícono de la aplicación	48
Figura 34. Interfaz para la selección de imágenes	48
Figura 35. Interfaz de las carpetas que contiene las imágenes procesadas	49
Figura 36. Interfaz de las imágenes almacenadas	49
Figura 37. Interfaz que muestra la imagen seleccionada	50
Figura 38. Interfaz donde muestra los resultados.....	50

1. INFORMACIÓN GENERAL

TÍTULO DEL PROYECTO:

“Desarrollo de un prototipo para el reconocimiento de plagas en plantaciones de babaco bajo invernadero utilizando redes neuronales”

FECHA DE INICIO:

5 de Abril 2021

FECHA DE FINALIZACIÓN:

5 de Agosto 2021

LUGAR DE EJECUCIÓN:

Invernaderos de la Provincia de Tungurahua

UNIDAD ACADÉMICA QUE AUSPICIA:

Ciencias de la Ingeniería y Aplicada

CARRERA QUE AUSPICIA:

Ingeniería en Informática y Sistemas Computacionales

PROYECTO DE INVESTIGACIÓN VINCULADO:

Inteligencia artificial e Inteligencia de Negocio.

EQUIPO DE TRABAJO:

Ing. Msc. Villa Quishpe William Manuel

Gamboa Murillo Patricio Alejandro

López Miniguano Isaac Samuel

COORDINADOR:

Nombre: Ing. William Manuel Villa Quishpe

Nacionalidad: Ecuatoriana

Residencia: Píllaro – Ecuador

Email: Manuel.villa@utc.edu.ec

Celular: 0983855980

Títulos obtenidos:

Título	Institución de Educación Superior
Diploma superior en comercio exterior	Universidad Regional Autónoma de los Andes
Magister en Interconexión de Redes	Escuela Superior Politécnica de Chimborazo
Licenciado en Sistemas Computacionales	Universidad Regional Autónoma de los Andes
Ingeniería en Sistemas e Información	Universidad Regional Autónoma de los Andes
Técnico Ejecutivo Analista de Sistemas	Universidad Regional Autónoma de los Andes
Tecnólogo en Computación e Informática	Universidad Regional Autónoma de los Andes

Tabla 1. Títulos obtenidos del coordinador

ESTUDIANTES:

Nombre: Gamboa Murillo Patricio Alejandro

Nacionalidad: Ecuatoriana

Fecha de Nacimiento: 1 de Julio 1994

Estado Civil: Soltero

Residencia: Ambato

Correo: patricio.gamboa3869@utc.edu.ec

Celular: 0980267022

Nombre: López Miniguano Isaac Samuel

Nacionalidad: Ecuatoriana

Fecha de Nacimiento: 13 de Octubre de 1995

Residencia: Ambato

Correo: Isaac.lopez0819@utc.edu.ec

Celular: 0960155795

ÁREA DEL CONOCIMIENTO:

Información y Comunicación (TIC)

LÍNEA DE INVESTIGACIÓN:

Tecnología de la Información y Comunicación (TICs)

SUB LÍNEA DE INVESTIGACIÓN DE LA CARRERA:

Robótica e inteligencia artificial e Inteligencia de Negocio

2. INTRODUCCIÓN

El presente proyecto de investigación se realiza en los invernaderos del agricultor Héctor Mecías ubicado en el cantón Patate de la provincia de Tungurahua, en la cual se presenta la problemática de los cultivos del babaco. Tal razón el objetivo de la investigación tecnológica es desarrollar un prototipo de software para identificar el tipo de plagas que se encuentre en cada una de ellas, utilizando herramientas que nos ayuden a agilizar el desarrollo. Para el mismo se empleará las metodologías de investigación enmarcadas en el tipo de investigación descriptiva, bibliográfica, técnicas e instrumentos de recolección de datos como entrevistas y observación con la finalidad de comprender la problemática e identificar las necesidades principales. Para el desarrollo del proyecto se realiza las herramientas de software como Phyton, Google Colab, Tensorflow, Open Cv, Keras, también utilizará Android Studio el cual servirá para crear la interfaz donde se mostraron los resultados del entrenamiento de la red neuronal. Con el fin de ayudar al agricultor la identificación del tipo de plagas existentes dentro del cultivo del babaco, empleando las metodologías de desarrollo evolutivo para el diseño del prototipo vertical, obteniendo como resultado un esquema integral para la delimitación del prototipo vertical, además se analizó resultados de modelos de redes neuronales empleados en proyectos de visión artificial mediante procesos de revisión bibliográfica, lo cual

permitió seleccionar un modelo adecuado que permita el entrenamiento de la red neuronal utilizando imágenes reales de plagas en plantaciones de babaco. De este modo se puede concluir que por medio de una interfaz móvil se mostrará los resultados procesados mediante la interfaz móvil.

2.1. EL PROBLEMA

Los cultivos de babaco bajo invernadero ocupan una gran extensión de hectáreas, pues se siembra desde pequeñas áreas hasta grandes áreas, ya sea, para consumo familiar o para comercio exterior. Actualmente en el Ecuador en la provincia de Tungurahua es común encontrar cultivos de babaco bajo invernadero los cuales presentan problemas de ataque de plagas, que afectan en la producción de la planta ocasionando grandes daños en la economía de los agricultores ver anexo “”. Este estudio pretende mediante el desarrollo de un prototipo vertical, comprobar el nivel de efectividad alcanzada en la detección de plagas en plantas de babaco bajo invernadero, considerando el uso de redes neuronales para el efecto.

2.1.1. Situación Problemática

Cada día, los nuevos avances tecnológicos y científicos demuestran que la visión e ingenio del ser humano parecen no tener fin. La búsqueda por emular la inteligencia humana, en lo que se denomina “inteligencia artificial”, ha transformado la forma en que se ve el mundo, bajo la posibilidad de que esta característica humana pueda residir en artefactos y medios electrónicos, con inherentes mejoras en tiempos de respuesta y precisión. Son muchas las aplicaciones de la inteligencia artificial, debido a que es un modelo computacional que ha demostrado gran eficiencia en el ámbito experimental e investigativo. El uso de las redes neuronales artificiales se encuentra en el reconocimiento de voz y la visión artificial, así como en procesos predictivos de alta complejidad y confiabilidad, y en distintas áreas del conocimiento, incluyendo la agrícola.

Una plaga puede definirse como “cualquier especie, raza o biotipo vegetal o animal, o agente patógeno dañino para las plantas o productos vegetales” [1]. Su relevancia en el ámbito agrícola radica que afectan entre un 20% y 40% de la producción mundial, con las consecuentes pérdidas económicas de miles de millones de dólares al año [2]; además el agro genera el 80% de los alimentos que se consumen en el

mundo [3]. La biodiversidad agrícola se ha visto drásticamente mermada en medio siglo como consecuencia del cambio climático: casi un tercio de las especies conocidas están en peligro de desaparecer, y el incremento de las temperaturas ha dado origen, entre otros efectos, a la aparición de sin número de plagas que amenazan a la agricultura [4].

En el Ecuador, la agricultura tiene una importante participación en su economía: aportó con un 7.71% del Producto Interno Bruto (PIB) del año 2019, y además generó el 28,6% del empleo total [5]. Los estudios han concluido que las plagas provocan el 43% de pérdidas en cultivos de banano, maíz, papa, palma, arroz y caña en el país [6]. La problemática generada en torno al manejo adecuado de las plagas trasciende más allá de lo económico, hacia el cuidado de la salud de la población. Tradicionalmente, se han venido empleando compuestos químicos como insecticidas, fungicidas y nematocidas para diezmar las plagas, aun cuando se conoce que su uso excesivo impacta de forma negativa al suelo y a la salud humana [7].

La detección temprana es sin duda la mejor alternativa que permite al agricultor tomar medidas menos invasivas, como el uso de pesticidas orgánicos, evitando que se afecte el crecimiento y desarrollo de las plantas, y asegurando la calidad del producto y la salud de los consumidores. En este punto, la tecnología juega un papel trascendental, con la publicación de varias investigaciones enfocadas a la detección temprana de plagas en ciertos tipos de cultivos, como por ejemplo de papa [8], arándanos [9], y sandía [10], las cuales poseen en común el uso de redes neuronales y el procesamiento digital de imágenes. Sin embargo, cada cultivo posee una morfología diferente y es susceptible a ciertos tipos de plagas y enfermedades, con lo cual su tratamiento a nivel tecnológico debe ser también específico.

En el caso del cultivo de babaco, no se han encontrado estudios o investigaciones que hagan referencia a la detección de plagas en base a soluciones tecnológicas. Su importancia en el agro ecuatoriano radica en que existen alrededor de 180 hectáreas sembradas con esta fruta bajo invernadero y a cielo abierto, siendo Tungurahua la provincia que posee más cultivos de este tipo [11]. Las principales plagas que afecta este tipo de cultivo son: nemátodos, ácaros y áfidos o pulgón [12]. Considerando

los antecedentes antes planteados resulta relevante establecer o comprobar, mediante el desarrollo de un prototipo de software, el nivel de efectividad alcanzada en la detección de plagas en plantas de babaco bajo invernadero, considerando el uso de redes neuronales para el efecto. Cabe destacar que la plaga de nemátodos se produce a nivel de la raíz de la planta, razón por la cual se excluye del ámbito del proyecto.

2.1.2. Formulación del problema

¿De qué manera se puede contribuir para el reconocimiento de plagas en plantaciones de babaco bajo invernadero?

2.2. OBJETO Y CAMPO DE ACCION

2.2.1. Objeto

Reconocimiento de plagas ácaros y pulgones en plantaciones de babaco bajo invernadero.

2.2.2. Campo de acción

Desarrollo de un prototipo de software para el reconocimiento de plagas en plantaciones de babaco bajo invernadero utilizando redes neuronales.

2.3. BENEFICIARIOS

2.3.1. Beneficiarios Directos

Investigadores en el área agrícola del país.

Agricultores de babaco bajo invernadero.

2.3.2. Beneficiarios Indirectos

Población en general que consume el babaco.

2.4. JUSTIFICACION

La identificación temprana y efectiva de las plagas en el babaco permitirá reducir las pérdidas económicas del sector agrícola de la región, así como asegurar la calidad del producto con un menor impacto a la salud de los consumidores. Los avances en el desarrollo de hardware y métodos de aprendizaje han permitido que el uso de redes neuronales esté generando importantes resultados en las tareas de detección y clasificación [13]. Existen varias investigaciones que demuestran que es posible diseñar soluciones informáticas que integran aprendizaje profundo en el

tratamiento de imágenes, conjuntamente con hardware especializado, con la finalidad de apoyar y mejorar la detección temprana de plagas [14].

Dentro del presente proyecto se propone generar un prototipo de software que se enfoque en la primera fase de la arquitectura de solución para la detección de plagas en el contexto de estudio, considerando factores de viabilidad en torno a recursos económicos y acceso a un entorno real de pruebas y aplicación. De esta manera, el primer paso consistirá en diseñar la solución integral, para posteriormente centrar los procesos investigativos en la programación y entrenamiento de un modelo de aprendizaje profundo basado en redes neuronales y procesamiento digital de imágenes. Se pone de manifiesto entonces la necesidad de aplicar los conocimientos adquiridos durante la formación académica, así como aplicar procesos investigativos, cuyos resultados deberán ser validados científicamente con pruebas estadísticas, de forma que éstos sean confiables.

Para el desarrollo del prototipo de software se cuenta con el apoyo del agricultor Héctor Mecías del cantón Patate de la provincia de Tungurahua, el cual permitirá la captura u obtención de la información inicial para el estudio. Así también se cuenta con una amplia literatura e investigaciones que se considerarán como punto de partida para el análisis de alternativas y configuración del modelo de aprendizaje profundo a ser empleado. Se espera que la metodología implementada pueda servir como punto de partida para otras investigaciones relacionadas a este ámbito.

Los beneficiarios directos del proyecto son investigadores en el área agrícola del país, que puedan implementar posteriormente la solución integral diseñada; así también son beneficiarios directos los agricultores de babaco en invernadero, los cuales obtendrán de su uso una mejor producción, y evitarán pérdidas por plagas en sus plantaciones. Por otra parte, los beneficiarios indirectos son la población en general que consume esta fruta, ya que la detección temprana evitará el uso de químicos que afecten su salud.

2.5. HIPÓTESIS

Mediante el diseño de un prototipo de software se permitirá el reconocimiento de plagas en plantaciones de babaco bajo invernadero.

Hipótesis	Variables	Indicadores	Técnicas
Mediante el diseño de un prototipo de software se permitirá el reconocimiento de plagas en plantaciones de babaco bajo invernadero.	Variable Dependiente Efectividad en el reconocimiento	• Redes neuronales	• Pruebas de prototipo • Aplicación móvil
		• Reconocimiento manual (experto)	• Observación
	Variable Independiente Plagas en plantas de babaco bajo invernadero	• Tipo de plaga	• Observación

Tabla 2. Hipótesis

2.6. OBJETIVOS

2.6.1. Objetivo General

Desarrollar un prototipo de software para el reconocimiento de plagas en plantaciones de babaco bajo invernadero mediante el procesamiento de imágenes, utilizando redes neuronales.

2.6.2. Objetivos Específicos

- Diseñar un esquema y analizar los resultados de desempeño de modelos de redes neuronales empleados en proyectos de visión artificial, mediante procesos de revisión bibliográfica, para la selección de un modelo adecuado al contexto de estudio.
- Entrenar la red neuronal, utilizando imágenes de plagas en plantaciones de babaco bajo invernadero para determinar el tipo de plaga.
- Implementar una interfaz móvil la cual permita mostrar los resultados de las imágenes procesadas mediante el entrenamiento de la red neuronal.

2.7. SISTEMA DE TAREAS

OBJETIVOS ESPECIFICOS	ACTIVIDADES	RESULTADO DE LAS ACTIVIDADES	DESCRIPCIÓN (TÉCNICAS E INSTRUMENTOS)
Diseñar un esquema y analizar los resultados de desempeño de modelos de redes neuronales empleados en proyectos de visión artificial, mediante procesos de revisión bibliográfica, para la selección de un modelo adecuado al contexto de estudio.	Revisar fuentes bibliográficas que se basen en proyectos de visión artificial para seleccionar un modelo adecuado.	Establecer y verificar todas las fuentes confiables (libros, artículos, proyectos relacionados), el cual nos permite seleccionar de manera eficaz el modelo a entrenar para el desarrollo de la red neuronal.	Análisis documental, ficha bibliográfica e imágenes.
Entrenar la red neuronal, utilizando imágenes de plagas en plantaciones de babaco bajo invernadero, para la realización pruebas de clasificación.	Selección y clasificación de imágenes por plagas para el entrenamiento de la red neuronal.	Determinar el tipo de plaga y el nivel de afectación producida en la hoja, para realizar la arquitectura correspondiente.	Trabajo de campo y toma de imágenes.
Implementar una interfaz móvil la cual permita mostrar los resultados de las imágenes procesadas mediante el entrenamiento de la red neuronal.	Desarrollo de la interfaz móvil.	Aplicación móvil	Aplicación móvil

Tabla 3. Sistema de tareas (planificación de actividades)

Elaborado por: Los Investigadores

3. FUNDAMENTACIÓN TEÓRICA

3.1. BABACO EN INVERNADERO

El babaco es una fruta que proviene de los valles andinos del Ecuador. Su crecimiento se da en ámbito natural, para el desarrollo de esta fruta se debe estar en climas templados los cuales oscilan entre los 1.500 y 2.500 metros sobre el nivel del mar [15].

La producción de esta fruta va tomando cada vez más fuerza y cada día se vuelve más favorable para las personas que se dedican a este tipo de cultivos, según estudios e investigaciones realizadas mediante la toma de datos se estima que la superficie sembrada con esta fruta puede alcanzar de entre 175 a 250 hectáreas de sembrío, donde la gran mayoría con el cultivo de esta fruta se da puede encontrar bajo invernadero [15].



Figura 1. Planta de babaco

3.1.1. Clasificación botánica

Su nombre científico es *Carica pentagona* (Heilb), pertenece a la familia Caricáceas, a la que pertenecen también el chamburo, toronche, jigacho y siglalón [15].

3.1.2. Morfología

➤ Raíz

Las raíces principales son gruesas y de consistencia carnosa que logran poca profundidad (un metro, o menos). Las raíces secundarias son largas, de 50 a 60 cm de longitud, también de consistencia carnosa de color café oscuro. Las raíces terciarias son de mayor proporción y ramificadas, un poco más blandas, débiles, quebradizas y tienen gran capacidad de retención de agua [15].

➤ Tallo

La planta de babaco es un arbusto con tallos semileñosos. El tronco es recto, cilíndrico, no tiene consistencia leñosa; cuando es joven es de color verde claro; de adulto toma un color marrón gris; su consistencia es fibrosa, esponjosa, puede alcanzar una altura entre 2 y 2,5 metros en un año [15].

➤ Hojas

Se encuentran insertadas a lo largo del tronco; son de color verde, por haber permanecido un buen período adheridas al tallo; tienen un largo pecíolo que, en algunos casos, pueden llegar hasta los 30 centímetros y es carnoso [15].

➤ Flores

Aparecen de manera continua en las axilas de las hojas, de forma acampanada, solitarias, de pétalos blanco-amarillento-verdoso; el cáliz es de color verde oscuro. El babaco es una planta monoica porque está constituida de flores exclusivamente femeninas, por lo tanto, no presenta anteras y filamentos [15].

➤ Fruto

Es una baya sin semilla, no necesita polinización para desarrollarse; es alargado de sección pentagonal; mediano, de unos 25cm de largo por 11 cm de diámetro. Los frutos grandes miden 12,5 cm de largo y 10,5 cm de diámetro; pesan 0,68 a 1,50 kg [15].

3.1.3. Propagación

La propagación se hace exclusivamente en forma vegetativa, empleando estacas maduras, sin hojas. Previo al enraizamiento es necesario dejar las estacas expuestas al aire en un lugar sombrío, durante 10 a 12 días, para que el corte cicatrice [16].



Figura 2. Propagación del babaco

3.2. PLAGAS EN EL BABACO

En las plantaciones de babaco se puede identificar 3 tipos de plagas las cuales son ácaros, pulgones y nemátodos, ver el anexo 3.

3.2.1. Pulgón (*Aphis* sp)

Son pequeños insectos de color negro, verde o gris, de acuerdo a la especie; se ubican en los brotes tiernos y en el envés de las hojas, succionan la sabia y transmiten virus [15].

3.2.2. Ácaros (*Tetranychus urticae*)

Son pequeñas arañas que clavan los estiletes en los frutos y absorben los jugos celulares; al vaciar las células, el tejido afectado adquiere una coloración marrón con el paso del tiempo provoca que el fruto se pudra [15].

3.2.3. Nemátodos (*Melodogine incognita*)

Los nemátodos producen nodulaciones en la raíz.

3.3. REDES NEURONALES

Las redes neuronales son en sí redes interconectadas en gran cantidad o que a su vez están en paralelo de elementos simples (usualmente adaptativos) y se encuentran organizados de forma jerárquica, las cuales intentan simular e

interactuar con los objetos del mundo real, de la misma forma en que trabaja el sistema nervioso biológico [17].

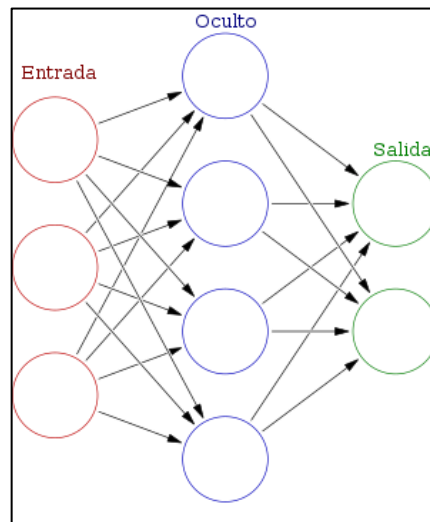


Figura 3. Red neuronal artificial

3.3.1. Ventajas que ofrece la red neuronal

Debido a su representación la forma en que las redes neuronales están constituidas y a los fundamentos que estas presentan simulan en gran número a la red del cerebro.

- Aprendizaje adaptativo. Es la capacidad o la forma de aprender para realizar tareas basadas en aprendizaje por medio de entrenamiento o en una experiencia inicial [17].
- Autoorganización. Una red neuronal está en la capacidad de crear su propia organización para procesar o representar la información que recibe mediante una etapa de entrenamiento (aprendizaje) [17].
- Tolerancia a fallos. Cuando una red se destruye parcialmente conduce a una degradación de su estructura; sin embargo, esta en la capacidad de retener la información, incluso sufriendo grandes daños en su estructura [17].
- Operación en tiempo real. Para que la red neuronal pueda trabajar los cálculos neuronales se deben realizar en paralelo; de tal manera se diseñan y fabrican máquinas con hardware especial para obtener la mayor capacidad [17].

- Fácil inserción dentro de la tecnología existente. Se pueden conseguir chips que estén fabricados y especializados para redes neuronales con el fin de mejorar su capacidad en ciertas tareas. Ello facilitará la integración modular en los sistemas existentes [17].

3.3.2. Elementos básicos que componen una red neuronal

En la figura 4 se puede apreciar un esquema de una red neuronal:

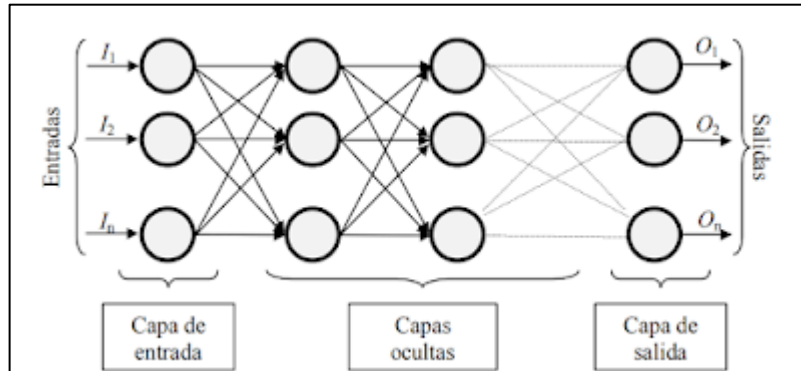


Figura 4. Red neuronal totalmente conectada

La misma que se encuentra constituida por neuronas que van interconectadas entre sí y formadas a través de tres capas (esto último puede variar). Los datos ingresan por medio de la “capa de entrada” pasan a través de la “capa oculta” y salen por la “capa de salida” [17].

Para un mejor entendimiento sobre el estudio de las redes neuronales, se debe conocer y aprender algo sobre cómo trabaja y como es el desempeño de las neuronas para saber cómo se emplean dentro de una red neuronal. En la figura 5 se hace una comparación entre la neurona biológica y la neurona artificial por ende se puede apreciar la gran similitud que existe en ambas.

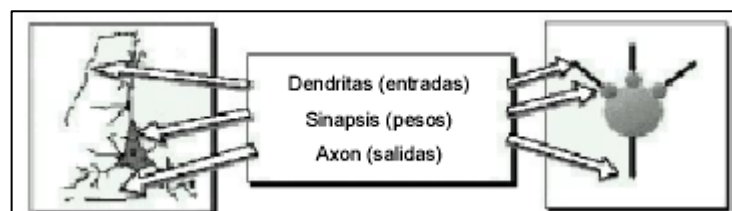


Figura 5. Comparación de una neurona biológica y una artificial.

Mientras una neurona puede presentarse de una forma muy pequeña en sí misma, al momento que estas se combinan cientos, miles o millones de ellas pueden estar en la capacidad de resolver problemas muy grandes y muy complejos. Dado un ejemplo el cerebro humano se compone de billones de neuronas [17].

➤ **Función de Entrada**

La neurona está en la capacidad de tratar a muchos valores de entrada y determinarlo como si fuera un solo dato; a este procedimiento se le da el nombre de entrada global [17].

➤ **Función de activación**

En esta función una neurona biológica puede estar activa (excitada) o inactiva (no excitada); es decir, que en cualquier momento pueden entrar en un “estado de activación”. En el ámbito de las neuronas artificiales también se presentan diferentes estados de activación; algunas de ellas pueden presentar solamente dos, al igual que presenta en las neuronas biológicas, pero otras pueden tomar diferentes valores dentro de un conjunto determinado [17].

➤ **Función de salida**

El valor que da como resultado de esta función es la salida de los datos procesados de la neurona, por ende, la función de salida es la que determina que valor se transfiere a las neuronas ya vinculadas [17].

3.3.3. Aprendizaje, Validación y Codificación

3.3.3.1. Niveles o capas de una red neuronal

La distribución de neuronas dentro de la red se realiza formando niveles o capas, con un número determinado de dichas neuronas en cada una de ellas. A partir de su situación dentro de la red, se pueden distinguir tres tipos de capas [17]:

- De entrada: es la capa encargada de recoger directamente la información que proviene de fuentes externas de la red.
- Ocultas: no tienen contacto con el exterior ya que son internas a la red. Este dado número de niveles puede estar en cero o en un numero extenso donde

las neuronas pueden estar interconectadas de distintas maneras lo que determina las distintas topologías de redes neuronales.

- De salidas: transfieren la información de la red al mundo exterior.

3.3.3.2. Tipos de neuronas artificiales

Se pueden clasificar acorde a los valores que estas pueden tomar. Por el momento se hace enfoque a las dos principales:

- Neuronas binarias.
- Neuronas Reales.

Las neuronas binarias solamente pueden tomar valores dentro del intervalo $\{0, 1\}$ o $\{-1, 1\}$, mientras que las neuronas reales pueden hacerlo dentro del rango $\{0, 1\}$ o $\{-1, 1\}$. Los pesos normalmente están restringidos a un cierto intervalo, aunque para aplicaciones específicas puede ser esto necesario [17].

3.3.3.3. Técnicas de decisión

La técnica o proceso de decisión se puede caracterizar como se muestra en el diagrama de la figura “6”

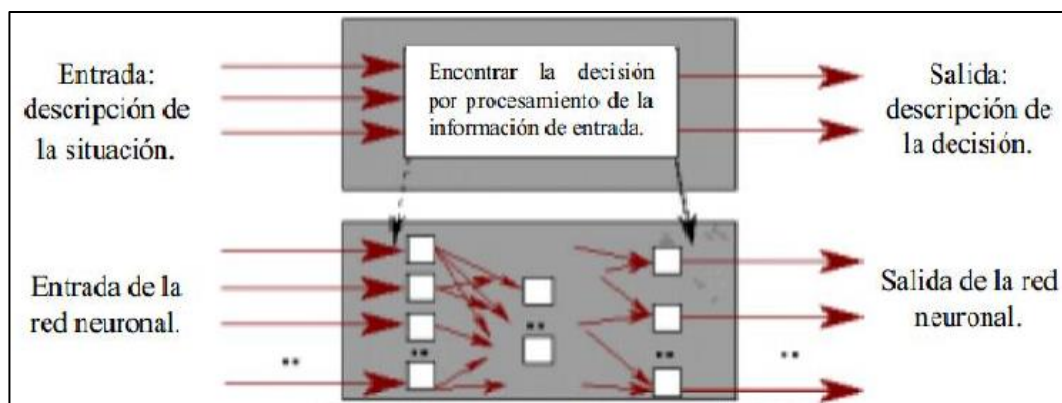


Figura 6. Esquema proceso de decisión

Lo mismo ocurre al utilizar una red neuronal como un sostén para la toma de decisiones. La salida de la red neuronal es directa o indirectamente la solución al problema o a la decisión a tomar [16].

3.3.3.4. Mecanismos de aprendizaje

En la función de entrada se procesan los datos a través de la red neuronal con el propósito de obtener datos de salida. También se habla que las redes neuronales

trabajan por medio de la extracción de generalizaciones desde un conjunto de datos determinado, dado en ejemplos anteriores de tales problemas de decisión.

Una red neuronal está en la capacidad de aprender a realizar cálculos de salida que sean correctos para cada arreglo o vector que se da en la entrada o en el conjunto de ejemplos, a este proceso se le da el nombre de entrenamiento o acondicionamiento. El conjunto de datos al que este proceso se basa es, por ende, llamado: conjunto de datos de entrenamiento [16].

Existen dos métodos de aprendizaje muy importantes que puede distinguirse:

- Aprendizaje supervisado: aprendizaje online.
- Aprendizaje no supervisado: aprendizaje offline.

Existe otro tipo de razonamiento o criterio con el que se puede clasificar las reglas de aprendizaje el cual se basa en ver si la red tiene la capacidad de aprender durante su funcionamiento tradicional o si el aprendizaje supone la desconexión total de la red, es decir, se inhabilita hasta que dicho proceso culmine.

➤ **Aprendizaje supervisado**

El aprendizaje supervisado se identifica porque tiene la capacidad de realizar el proceso mediante un entrenamiento bajo la supervisión de un agente externo que es quien determina la respuesta que debería generar la red a partir de una entrada determinada [16].

El supervisor es el encargado de controlar la salida de los datos de la red y si el resultado no coincide con lo deseado, se procederá a modificar los datos de los pesos de las condiciones, de tal manera que así se pueda obtener que la salida sea lo más semejante a la deseada.

En este tipo de aprendizaje por lo general se puede considerar tres formas de llevarlo a cabo, dando lugar a los siguientes aprendizajes supervisados:

- Aprendizaje por corrección de error. Consiste en ir ajustando los pesos de las conexiones de la red acorde a la diferencia entre los valores deseados y los que se obtuvieron a la salida de la red, es decir, en función del error cometido en la salida [16].

- Aprendizaje por refuerzo. Se trata también de un aprendizaje controlado, pero que es más lento que el anterior, el cual está basado en la idea de no disponer de un desarrollo completo del comportamiento deseado, es decir, los resultados no indican los pesos del entrenamiento exactamente a la salida que se desea que proporcione la red ante una determinada entrada [16].
- Aprendizaje estocástico. Este aprendizaje esta constituido básicamente en realizar cambios aleatorios en los valores de los pesos de las conexiones de la red.
- **Aprendizaje no supervisado**

Las redes de aprendizaje no supervisado no requieren influencia externa para ajustar los pesos de las conexiones entre sus neuronas. La red no recibe ninguna información por parte del entorno que le indique si la salida generada es respuesta a una determinada entrada es o no es correcta [16].

En cuanto a los algoritmos de aprendizaje no supervisado, en general se suelen considerar dos tipos, que dan lugar a los siguientes aprendizajes:

- Aprendizaje hebbiano. Esta regla de aprendizaje es la base de muchas otras, la cual pretende medir la familiaridad o extraer características de los datos de entrada [16].
- Aprendizaje competitivo y comparativo. Se orienta a la clusterización o clasificación de los datos de entrada. Como característica principal del aprendizaje competitivo se puede decir que, si un patrón nuevo se determina que pertenece a una clase reconocida previamente, entonces la inclusión de este nuevo patrón a esta clase matizará la representación de la misma [16].

3.3.4. Principales Topologías

La topología o arquitectura que está constituida una red neuronal se presenta en la organización y disposición de las neuronas en la misma, la cual forma capas o agrupaciones de neuronas que se encuentran alejadas de la entrada y de la salida de dicha red [17].

Dado esto se puede concluir que los principales parámetros a tomar en cuenta de una red son: número de capas, número de neuronas por capa, grado de conectividad y el tipo de conexión que tenga entre neuronas.

- Redes monocapa. Tienen la capacidad de establecer conexiones entre las neuronas que se presentan en la única capa está constituida por la red. Las redes mono capas son usadas generalmente en tareas que se relacionan con lo que se denomina como auto asociación (regenerar información de entrada que se presenta a la red de forma incompleta o distorsionada) [16].
- Redes multicapa. Son aquellas que están formadas por un conjunto de neuronas agrupadas por varios niveles de capas [16].

En este caso para poder distinguir una capa la cual pertenece a una neurona hay que tomar en cuenta las señales que entran al inicio y el destino final de la señal de salida.

- Conexión entre neuronas. La conectividad que se realiza entre los nodos de una red está asociada con la forma en que las salidas de las neuronas están relacionadas para convertirse en entradas de otras neuronas [16].
- Redes de propagación hacia atrás. El nombre de backpropagation resulta de la forma en que el error es propagado hacia atrás de la red neuronal, en otras palabras, el error se propaga hacia atrás desde la capa de salida [16].

Dado esto permite que los pesos que se dan sobre las conexiones de las neuronas que se ubican en las capas ocultas varíen durante el entrenamiento.

3.3.5. Redes neuronales convolucionales

Una red convolucional viene hacer un tipo de red multicapa la cual está constituida de diversas capas convolucionales y de pooling que se presenta tipo submuestreo netamente alternadas entre sí, y al final presenta una serie de capas full-connected como una red perceptrón multicapa [18].

“Las redes neuronales convolucionales (CNN) son un tipo particular de red neuronal las cuales están inspiradas en el funcionamiento de la corteza visual del cerebro. Estas redes están diseñadas con la finalidad de poder resolver problemas de visión artificial como el reconocimiento de patrones, aunque pueden tener otros usos como la clasificación de textos o el procesamiento de lenguaje natural” [19].

3.4. VISIÓN ARTIFICIAL

La visión artificial o en su representación de la visión por computador trata o intenta emular la capacidad de algunos seres vivos para ver una escena o imagen, tratar de entenderla y actuar en consecuencia [18].

Es un sistema de visión que fue establecido con el fin de automatizar los procesos de percepción visual por medio de imágenes digitales. La visión artificial puede ser definida como los procesos de caracterización u obtención para la interpretación de información de imágenes obtenidas desde un mundo tridimensional a partir de imágenes bidimensionales [18].

Estos procesos pueden presentar subdivididos en seis procesos como se muestra en la figura 7.

Procesos	Nivel de Visión	Entrada	Salida	Área
1. Captura	Bajo	Imagen	Imagen	Procesamiento de imágenes
2. Pre-procesamiento				
3. Segmentación	Medio	Imagen	Grupos de píxeles en bruto (objetos o regiones)	Análisis de Imágenes
4. Descripción		Objetos o regiones	Información cuantitativa de los objetos o regiones	
5. Reconocimiento (clasificación)		Información cuantitativa	Objetos clasificados en categorías	
6. Interpretación	Alto	Objetos clasificados en categorías	Comprensión de la escena	Visión por Computador

Figura 7. Procesos de la visión artificial

3.4.1. Componentes de la visión artificial

Los elementos necesarios que se usan en un sistema de visión artificial son:

Sensor óptico. El sensor puede ser una cámara color o monocromo que produce una imagen completa del dominio del problema cada 1/30 segundos [17].

Tarjeta de adquisición de imagen. Permite digitalizar la señal de video entregada por el subsistema anterior [17].

Computador. Una vez digitalizada la imagen, debe ser almacenada en la memoria de un computador para su posterior procesamiento y manipulación por programa [17].

Monitor de video. Permite visualizar tanto las imágenes o escenas captadas como los resultados de procesamiento de dichas imágenes [17].

En la figura 8 se puede apreciar la iluminación, la cámara, y el proceso de digitalización que permite visualizar la imagen digital y por otro lado la extracción de parámetros para su posterior proceso de análisis y control que actúa sobre el entorno del objeto.

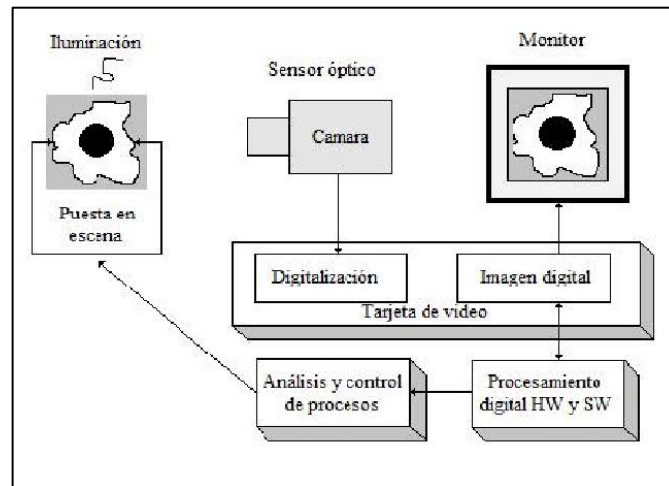


Figura 8. Diagrama de bloques de un sistema SVA

3.5. PHYTON

Phyton fue desarrollado como un lenguaje de programación de alto nivel, interpretado y multipropósito. En los últimos años el uso de este lenguaje ha ido creciendo significativamente y en la actualidad es uno de los lenguajes de programación más empleado para el desarrollo de software [18]. Totalmente de acuerdo ya que hay varios mitos alrededor de Python lo cual se basa en su pobre rendimiento lo cual no es para nada correcto, ya que, al ser desarrollado como un lenguaje interpretado estos por lo general tienden a ser un poco más lentos que los lenguajes compilados, Python con la diferencia de otros lenguajes interpretados, cuenta con toda su librería estándar basada en el lenguaje de programación en C, lo cual lo convierte en que sus funciones primitivas sean bastante eficientes. Además, puede presentar su compilación de código a bytecodes, similar al que usan Java y .NET, lo que hace que la optimización sea aún rápida para el proceso de interpretación. Otra de las características más importantes y principales de Python es la capacidad de inclusión sobre un sistema de documentación. La documentación

fue primordial a la hora de tomar en cuenta la hora y el diseño del lenguaje. Dado un ejemplo, todas las clases y funciones tienen una variable implícita que se llama docstring que es la cadena de documentación. En estas cadenas se describe el uso de esa función o clases y los parámetros que intervienen en detalle. La sintaxis de estos docstring es muy similar a la de JavaDoc.

3.6. GOOGLE COLAB

Es un entorno colaborativo de Google el cual permite trabajar con Notebooks además del lenguaje de programación de Phyton y que a su vez permite almacenar dichos cuadernos y trabajar con datos almacenados en drive y así se puede compartir con un equipo de trabajo [19].

Desde Colab se puede trabajar con ejemplos que van desde la práctica en la ciencia de datos con las librerías numpy y Matplotlib al aprendizaje automático con TensorFlow [19].

3.7. TENSORFLOW

Es una librería desarrollada por Google con la capacidad de construir y entrenar redes neuronales. Se encuentra influenciada por la librería Theano, la cual es la predecesora de DistBelief (librería creada por Google) y se hace uso tanto para la investigación como para la producción de los propios productos de Google [20].

3.8. OPEN CV

Es una biblioteca de código abierto que contiene implementaciones que abarcan más de 2500 algoritmos. Además, está especializada en el sistema de visión artificial y machine learning [21].

Fue diseñado con el fin de ser muy eficiente y con gran enfoque en aplicaciones que se dan en tiempo real, es una librería bastante conocida y ha sido usada en varias aplicaciones como por ejemplo sistemas de reconocimiento de objetos y sistemas de seguridad.

3.9. KERAS

Es una biblioteca usada en redes neuronales artificiales la cual es de código abierto. Esta desarrollada bajo el lenguaje Phyton y puede ser ejecutada sobre diferentes plataformas ya sea TensorFlow o Theano. Keras está diseñado para ir trabajando

por bloques de la arquitectura en que se basa cada red neuronal, incluyendo redes convolucionales y recurrentes, que son las que permiten, junto a los bloques “más tradicionales”, entrenar modelos Deep learning [22].

3.10. REMOVEBG

Es un editor de fotos online que permite quitar el fondo a las fotos con el fin de crear imágenes de buena calidad y con el fondo transparente, para que se pueda usar en trabajos o ya sea para guardar diseños increíbles.

3.11. GOOGLE RESNET 50

ResNet50 permite entrenar redes muy profundas de más de 100 capas es el desarrollo más crítico de una estructura, en los últimos años una gran cantidad de conocimiento tiene una estructura residual, su estructura es simple y efectiva, y se beneficia de su amplia aplicación [24].

3.12. ANDROID STUDIO

“Android Studio es el entorno de desarrollo específico de Android, basado en IntelliJ IDEA (entorno de desarrollo conocido en la comunidad de desarrolladores Java). Aporta muchas novedades respecto a eclipse.” [24]. Totalmente de acuerdo ya que aporoto con la misma opinión, en que Android Studio es un entorno el cual nos permite trabajar con el lenguaje de programación Java, a la misma vez se puede trabajar con otro lenguaje de programación conocido como Kotlin el cual nos va a permitir también crear aplicaciones móviles. Algunas de las características de Android Studio son: brindar soporte en la construcción dentro de Gradle, existen plantillas para crear diseños comunes de Android y a la misma vez otros componentes.

3.13. APLICACIÓN MOVIL

Es un software el cual nos va permitir descargar, entonces a la misma vez nos permite acceder directamente desde cualquier teléfono u otro dispositivo móvil que se pueda utilizar aplicaciones que sean sustentables a la programación de aplicación móvil. Para lo cual se necesita algún aparato móvil el cual contenga internet, el cual también debemos saber el tipo de sistema operativo móvil, puede ser: Android, Apple, Microsoft [24].

4. MATERIALES Y MÉTODOS

4.1. Experimental

Este método de investigación nos permite tener un máximo control de las variables de entorno, ya que los investigadores actúan y observan, para constatar las consecuencias de esa actuación ya que los resultados obtenidos se basan a un proyecto o prototipo.

4.2. De Campo

Este tipo de investigación es utilizado al momento de la siembra de la planta de babaco, para así poder determinar las diferentes plagas de la misma, las cuales se registraron mediante fotografías.

4.3. Documental

Esta documentación es realizada al momento de las revisiones bibliográficas el cual nos permite determinar el alcance de daño de la plaga del babaco que se va centralizando en el análisis de las redes neuronales, el cual nos hará conocer el tipo de plaga que es.

4.4. Tecnológica

Esta técnica permite al investigador obtener información en base a una propuesta ya establecida la cual permita mejorar procesos y adaptarlos a la vida cotidiana, tomando en cuenta que los instrumentos de investigación sean los mismos y permitan una recolección de datos eficaz.

4.5. Métodos de Investigación

4.5.1. Método hipotético deductivo

Este método es considerado ya que permite conocer si las afirmaciones hipotéticas realizadas son falsas o son verdaderas esto se lo hace al momento de poner en práctica el proyecto de investigación. Al introducir en el modelo de entrenamiento se obtendrá un análisis de la fotografía en el cual se observará el tipo de plaga que existe.

4.6. Técnicas de Investigación

En base a los tipos de investigación tenemos varios, que nos permitirá recolectar la información para establecer la elaboración del prototipo de software, en donde tenemos la entrevista como medio de investigación más recurrente y utilizado por los investigadores.

4.6.1. Entrevista

Como grupo investigador hemos usado la técnica de la entrevista ya que esto nos permite conocer todo lo relacionado a las plantaciones y plagas del babaco.

4.6.2. Observación

Esta técnica nos permitió analizar un objeto con el fin de obtener la información necesaria para la investigación realizada.

4.7. Instrumentos de investigación

Aquí se estableció distintos parámetros e instrumentos con la finalidad de conocer todo lo relacionado a las plantaciones y plagas del babaco.

4.7.1. Ficha bibliográfica

Se hizo uso de este instrumento de investigación con el fin de registrar la bibliografía consultada que permitieron escoger las ideas primordiales y escoger la técnica de procesamiento digital de imágenes para el entrenamiento de la red neuronal.

4.7.2. Fotografía

Esta técnica es utilizada para la recolección de imágenes que el modelo Deep learning requiere para las respectivas pruebas y entrenamiento de la red neuronal.

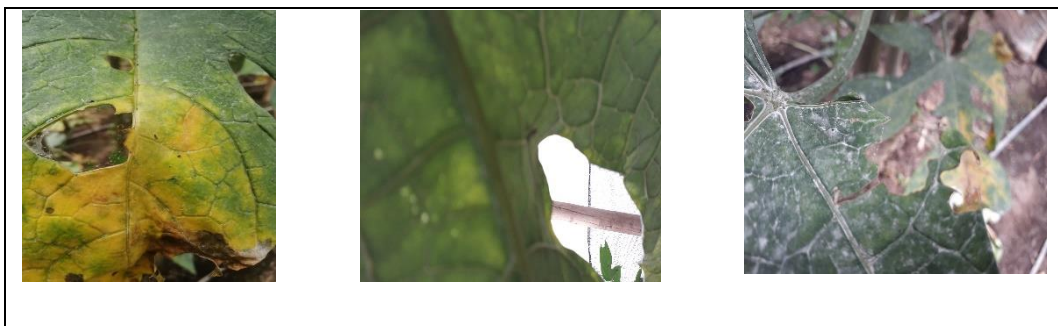


Figura 9. Fotografías tomadas por el grupo de investigadores.

4.8. Metodologías de desarrollo de software

Para el desarrollo de esta interfaz se realizó un análisis en base a proyectos de inteligencia artificial implementados en otras áreas agrícolas, el cual permitió establecer una revisión bibliográfica. Y así delimitar el proyecto a desarrollar.

4.8.1. Modelo de desarrollo evolutivo

El equipo de desarrollo consideró pertinente utilizar la metodología de desarrollo evolutivo ya que permite expandir incrementos de un producto de software operacional donde la dirección de la evolución la dicta la experiencia con el sistema. Los cambios a menudo suelen corromper la estructura del software. El desarrollo evolutivo solo es recomendado para pequeños y medianos sistemas.

4.9. Protocolo para la clasificación y procesamiento digital de imágenes

4.9.1. Selección y captura de imágenes

Las imágenes de las hojas y el fruto de babaco fueron capturadas manualmente obteniendo como resultado un total de 2500 fotografías.

4.9.2. Preprocesamiento

Con las imágenes ya obtenidas por parte del grupo investigador se entrenó la red neuronal para las respectivas pruebas de clasificación.

4.9.3. Procesamiento

A nivel general este proceso facilitó la selección y clasificación de las mismas obteniendo como resultado cincuenta imágenes por cada categoría.

5. ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS

5.1. Herramientas de programación

Se utilizó las librerías Open CV, Removebg, Google Colab para el tratamiento de imágenes además se usó Keras, Google resnet 50, tensorflow, bajo el lenguaje de programación Python, A la vez para crear la interfaz móvil se utilizó Android Studio (bajo lenguaje de programación Java) la razón del uso de estas herramientas es porque el proyecto hace uso del reconocimiento de imágenes, por lo que la interfaz va hacer desarrollada bajo estos lineamientos.

HERRAMIENTAS
Open CV
Google Colab
Pc
Removebg
Keras
Android Studio

Tabla 4. Herramientas de desarrollo

5.2. Seguimiento de la metodología de Desarrollo.

Para la construcción y la elaboración del proyecto se considera el modelo de desarrollo evolutivo ya que consta de una versión inicial que luego de ser expuesta se va refinando acorde a los nuevos requerimientos por parte del cliente o usuario con el fin de cumplir los lineamientos y procedimientos para llegar al producto final.

5.3. Estados de la metodología de desarrollo

Para levantar la información en cuanto a las plagas dentro de las plantaciones de babaco se estableció una información cualitativa y cuantitativa la cual permitió al grupo investigador conocer el tipo de afectación que se presenta en estas plantas.

Para la investigación se realizó una entrevista personal la cual conto con la presencia del agricultor Héctor Mecías dueño de la hacienda “El Porvenir”, y de los estudiantes investigadores. El cual permitió optar por el modelo de desarrollo evolutivo, ya que ello nos va ayudar en la construcción de versiones sucesivas y este nos permitirá detectar de mejor manera el tipo de plaga que se presenta en las plantaciones de babaco.

5.3.1. Especificación

En esta fase se determinó la problemática, los objetivos, la hipótesis y marco teórico para determinar el alcance del prototipo de software el cual consiste en el entrenamiento de la red neuronal que permita la detección de plagas.

5.3.2. Desarrollo

En esta fase se diseñó un esquema y se analizó resultados de modelos de redes neuronales empleados en proyectos de visión artificial mediante procesos de revisión bibliográfica, lo cual permitió seleccionar un modelo adecuado que permita

el entrenamiento de la red neuronal utilizando imágenes de plagas en plantaciones de babaco.

5.3.3. Validación

En esta fase se entrena la red neuronal con las imágenes ya clasificadas con el fin de obtener como resultados el tipo de plaga que se presenta en las hojas y frutos de babaco cuyo resultado será plasmado en una interfaz móvil.

5.4. Definición de Roles de Equipo

Nombre	Ing. Msc. Manuel Villa
Rol	Jefe de Equipo
Responsabilidad	Revisión del proyecto
Información contacto	Manuel.villa@utc.edu.ec

Tabla 5. Rol de equipo (equipo de trabajo)

Nombre	Patricio Alejandro Gamboa Murillo
Rol	Equipo de Trabajo
Responsabilidad	Investigar y generar el código de la interfaz
Información contacto	Patricio.gamboa3869@utc.edu.ec

Tabla 6. Rol de equipo (equipo de trabajo)

Nombre	Isaac Samuel López Miniguano
Rol	Equipo de trabajo
Responsabilidad	Investigar y generar el código de la interfaz
Información contacto	Isaac.lopez0819@utc.edu.ec

Tabla 7. Rol de equipo (equipo de trabajo)

5.7. Desarrollo del esquema integral

En la fase de desarrollo se ha obtenido como resultado el diagrama o esquema principal que sirve para delimitar el alcance del prototipo de software desarrollado, el cual consiste en la primera fase que es programación y entrenamiento de modelo Deep learning.

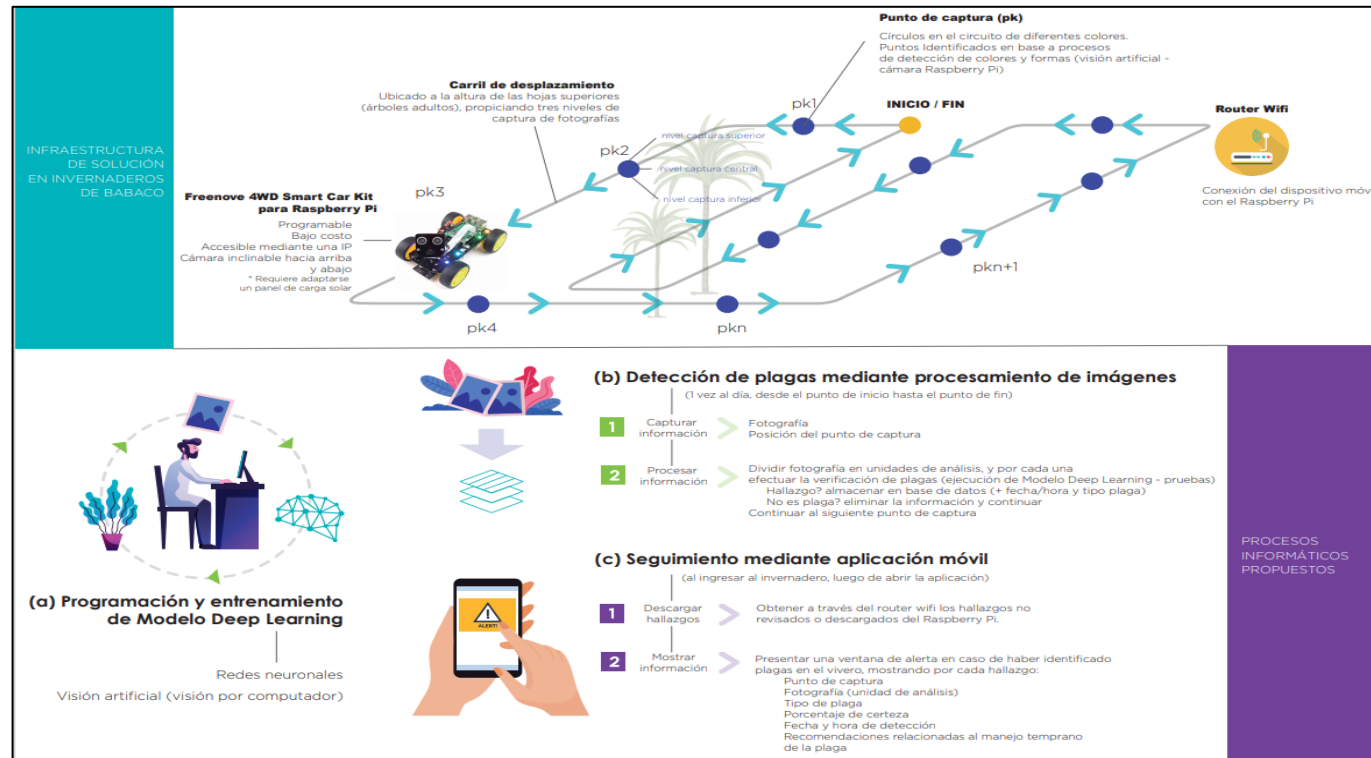


Figura 10. Diagrama del modelo principal.

5.8. Desempeño de redes neuronales en proyectos de visión artificial

En esta fase de desarrollo se obtiene como resultado que el proceso de análisis de desempeño de redes neuronales en proyectos de visión artificial, tuvo como objetivo principal la selección de un modelo adecuado al problema de investigación, para lo cual se desarrolló un proceso de revisión bibliográfica. Se empleó el buscador de Google académico, con los siguientes términos y parámetros de exploración:

- *Términos de búsqueda:* "redes neuronales" and "modelo" and "plagas" and "visión artificial".
- *Intervalo de tiempo:* 2017 – 2021 (5 años)
- *Ordenamiento de resultados:* por relevancia

Se obtuvieron un total de 83 resultados, los mismos que fueron revisados minuciosamente, a fin de priorizar las investigaciones que se listan en la Tabla 8. Dentro del proceso de revisión, se descartaron investigaciones que cumplieron uno o varios de los siguientes criterios de exclusión:

- El objetivo principal de investigación no se centra en las variables del presente estudio (por ejemplo: investigaciones que solo refieren a las variables en su marco teórico conceptual, o estudios enfocados a la detección de elementos de otra naturaleza como figuras, humanos, y objetos).
- No se encuentra claramente descritos uno o varios de los siguientes aspectos de análisis: el modelo de red neuronal, o la configuración proceso de entrenamiento.
- Los resultados cualitativos o cuantitativos no se encuentran enmarcados bajo indicadores de eficiencia, efectividad, confiabilidad, exactitud, o similares.

De esta manera, se priorizaron aproximadamente el 11% de los resultados de búsqueda obtenidos.

5.8.1. Aspectos comunes entre las investigaciones

En cuanto a los aspectos recurrentes o comunes de la mayoría de investigaciones descritas en la tabla 8, puede establecerse que:

- Emplean Redes Neuronales Convolucionales (RNC), con aprendizaje profundo.
- Utilizan el Framework (plataforma de software para aprendizaje profundo) TensorFlow para el desarrollo y entrenamiento de los modelos de aprendizaje automático (Python como lenguaje de programación).
- Han calculado técnicamente la efectividad de las siguientes arquitecturas de aprendizaje profundo: AlexNet, GoogLeNet, Inception V3, ResNet 50 y MobileNet (versiones 2 y 5). Los mejores resultados de confiabilidad se observaron en las siguientes arquitecturas: GoogLeNet (94,47% y 99,72%), MobileNet (89%) y ResNet 50 (96,68%).
- Respecto a la arquitectura hardware para el entrenamiento del modelo, se asume la necesidad de contar con una computadora de altas prestaciones, o realizar un procesamiento en la nube con Google Colaboratory.

En relación al literal (c), surgió la necesidad de establecer un factor decisorio sobre la arquitectura de red de aprendizaje profundo a ser utilizada, entre las dos opciones con confiabilidad mayor al 90%: GoogLeNet y ResNet50. Por una parte, la arquitectura de GoogLeNet tiene 27 capas de profundidad (más de 100 capas si contamos las capas en el módulo de inicio por separado), y genera aproximadamente un 6,67% de tasa de error; ResNet50 maneja 152 capas de profundidad, y una tasa de error de aproximadamente el 3,57% (Shah, Bakrola, y Pati, 2019) (Bianco, Cadene, Celona y Napoletano, 2018). Considerando estos parámetros, se seleccionó a ResNet 50 como la arquitectura de red de aprendizaje profundo del presente estudio, de la cual también se ha verificado la existencia de una extensa documentación técnica y práctica para su aplicación.

Título de la investigación	Autores y año de publicación	Modelo de red neuronal	Configuración del proceso de entrenamiento	Arquitectura de entrenamiento del modelo	Resultados destacados
Redes neuronales artificiales para la detección de plagas en cultivo de jitomate bajo invernadero	Berra, Pérez, y Cuautle (2019)	Red neuronal artificial de convolución basada en Keras del framework TensorFlow (algoritmos en lenguaje Python).	150 fotografías de alta resolución por cada plaga. Entrenamiento con tres variantes a las que se denominaron “Keras A”, “Keras B” y “Keras C”: quince intentos de entrenamiento sobre la red neuronal artificial con un total de quinientas ochenta épocas (epocs) por cada intento.	Bajo el framework de Intel (acelerador Intel Neuronal Stick 2).	Cada variante generó aproximaciones de detección y clasificación de entre 45% y 90% de confiabilidad.
Sistema inteligente de reconocimiento de imágenes para apoyar el diagnóstico de plagas y enfermedades en el cultivo de arroz en el departamento de Lambayeque en el año 2019	Galán (2021)	Redes neuronales convolucionales, bajo el modelo secuencial de Keras.	15000 imágenes.	No definido	Se probó el reconocimiento preciso de las plagas, mediante la validación de expertos.
Detección de enfermedades y plagas en cultivos mediante Machine Learning	Pereyra (2020)	Red neuronal convolucional basada en TensorFlow (en plataformas cloud)	146 imágenes: 49 correspondientes cladosporium, 43 a oidio y las restantes 54 sin síntomas de enfermedad. Se realizó el aumento de las imágenes, generando 30 imágenes para cada imagen que poseía síntomas etiquetados, lo que resultó en un total de 2906 imágenes para el entrenamiento. El modelo se entrenó por un total de 80.000 etapas de entrenamiento mediante el	Google Colaboratory (compatible con Python): en su versión gratuita provee una GPU K80, 12GB de memoria RAM y 12 horas de uso continuo hasta que se reinicie el entorno; librerías pre instaladas y conectividad a Github y Google Drive	No posee información cuantitativa relacionada a la medición de la eficiencia o confiabilidad en la detección, solo manifiesta que “logró detectar de manera correcta los síntomas”

Título de la investigación	Autores y año de publicación	Modelo de red neuronal	Configuración del proceso de entrenamiento	Arquitectura de entrenamiento del modelo	Resultados destacados
			modelo pre entrenado "faster rcnn inception v2"		
Desarrollo de una red neuronal artificial para clasificar la severidad del Virus del Mosaico Dorado a nivel foliar en frijol (<i>Phaseolus vulgaris L.</i>)	Ardón y Morales (2020)	Dos topologías de redes neuronales convolucionales comparadas: GoogleNet y ResNet50, implementadas en TensorFlow	3409 imágenes, clasificadas en tres escalas de severidad. Las redes neuronales fueron entrenadas durante 40 ephocs. La capa de clasificación se entrenó durante 35 ephocs, y en los 5 ephocs restantes se utilizó fine-tuning, entrenando las capas cercanas a la de clasificación.	No definido	ResNet50 alcanzó una mejor exactitud (96,68%) en comparación a GoogLeNet (94,47%), cuando los datos se encuentran organizados en dos categorías (alta y baja) y las características se extraen desde capas intermedias.
Desarrollo de un sistema de detección y conteo automático para Metisa plana Walker (Lepidoptera: Psychidae)	Shariff, Ahmad, Aris, Halin, y Moslim (2019)	Redes neuronales convolucionales	RNC preentrenada.	No definido	Precisión de hasta 95%, con una mayor precisión a 30 cm de distancia.
Comparación de arquitecturas de redes neuronales convolucionales para la clasificación de enfermedades en tomate	Gutiérrez (2019)	Redes Neuronales convolucionales: AlexNet, GoogleNet, Inception V3, ResNet 18 y ResNet 50	RNC preentrenadas y técnica de fine-tuning,	No definido	GoogleNet logró un mayor desempeño en comparación a las demás, alcanzando un porcentaje de exactitud de 99.72%, con un error del 0.28%.
Detección de enfermedades foliares con arquitecturas de redes neuronales convolucionales	Huerta, González, Rodríguez, y Amabilis (2020)	Redes neuronales convolucionales (librería Keras). Se evaluaron cinco arquitecturas de red de deep learning diferentes:	Conjunto de datos ImageNet, el cual contiene 1,2 millones de imágenes de 1000 categorías.	Intel i7-7700 y tarjeta de video NVIDIA GTX1070 con 8GB en RAM.	VGG16: 90% de sensibilidad y 90% de exactitud (mejor evaluado)

Título de la investigación	Autores y año de publicación	Modelo de red neuronal	Configuración del proceso de entrenamiento	Arquitectura de entrenamiento del modelo	Resultados destacados
		VGG16, InceptionV3, MobileNetV2, Xception y ResNet50V2			
Predicción de agentes patógenos en plantas ornamentales utilizando redes neuronales	Escobar (2018)	Redes neuronales convolucionales: se utilizó y reentrenó AlexNet, una CNN reconocida y empleada en otros proyectos.	75 imágenes en total para dos categorías: hojas dañadas y hojas sanas. 100 epochs y 300 iteraciones por entrenamiento.	Intel i7-7700 y tarjeta de video NVIDIA GTX1080TI 11GB GDDR5X, y RAM 16GB DDR4	En el primer entrenamiento se observa el 46% y 53% de efectividad para la predicción. En el tercer entrenamiento, la red obtuvo el 100% de efectividad en la predicción
Identificación de síntomas de Huanglongbing en hojas de cítricos mediante técnicas de deep learning	Berger, Preussler, y Agostini (2018)	Redes neuronales convolucionales, arquitecturas evaluadas: Inception versión 3 y MobileNet versión 0.5 (utilizando Tensorflow y Python)	1200 imágenes se dividieron en los conjuntos de entrenamiento y evaluación. Entrenamiento: 800 imágenes divididas en cuatro carpetas rotuladas como Asintomática, Magnesio, Zinc y HLB. Evaluación: 400 imágenes restantes, 100 en cada grupo. 500 iteraciones	Notebook con procesador i5-3230M de 2.6 GHz, 4 GB de RAM DDR3, SSD 240 GB y Windows 10 64 bits.	Clasificó correctamente el 89% de las imágenes de hojas utilizando el modelo MobileNet (mejor evaluado)

Tabla 8. Resultado de la investigación modelos de redes neuronales empleados en proyectos de visión artificial.

Nota. Basado en procesos de revisión bibliográfica de los autores.

5.8.2. Aspectos no comunes entre las investigaciones

Respecto a la configuración del proceso de entrenamiento, no se ha identificado una tendencia o enfoque claro sobre la cantidad de imágenes que deben ser empleadas. En su mayoría, los investigadores generaron sus propios datasets con cantidades de fotos en un rango entre 75 a 3409 imágenes; Huerta, González, Rodríguez, y Amabilis (2020) emplearon datasets predefinidos (de ImageNet) con millones de imágenes. En cuanto al aumento o argumentación de las fotografías, la mayoría de autores emplearon entre 40 y 580 epochs o etapas; por otra parte, Pereyra (2020) configuró el entrenamiento para 80.000 etapas.

5.8.3. Aspectos comunes entre las investigaciones

La mayoría de estudios efectúan un análisis respecto a una categorización no mayor a 4 factores (relacionados a los niveles de severidad del virus o tipo de plaga/enfermedad), lo cual responde con la categorización o tipos de plagas del babaco y que se mencionaron en apartados anteriores (áfidos y ácaros). Escobar (2018) incluye dentro de su categorización a las “hojas sanas”, con la finalidad de evitar errores de clasificación.

En este punto resulta relevante describir ciertas particularidades de las plagas de babaco, identificadas mediante el proceso de observación:

- A los frutos de la planta les afectan solo un tipo de plaga: ácaros.
- A las hojas de la planta les afectan ambos tipos de plaga: áfidos o pulgones, y ácaros.
- Las hojas de la planta, por lo general, se muestran afectadas por ambos tipos de plaga a la vez, razón por la cual la obtención de suficientes fotografías asociadas a una sola plaga no fue posible.

La categorización de la presente investigación, por lo tanto, se planteó como:

- *hojas_acapul*: Hoja con presencia de plaga (ácaros o pulgones/áfidos)
- *hojas_sanas*: Hoja sin presencia de plaga
- *frutos_ácaro*: Fruto con presencia de plaga (ácaros)
- *frutos_sano*: Fruto con presencia de plaga (ácaros)

5.8.4. Modelo de red neuronal adecuado al problema de investigación

En base a los antecedentes expuestos, el modelo de red neuronal seleccionado para la presente investigación, cumple los siguientes parámetros:

- Red neuronal convolucional, con aprendizaje profundo.
- Framework TensorFlow con Keras para el desarrollo y entrenamiento de los modelos de aprendizaje automático (Google Colab y Python como lenguaje de programación).
- Algoritmos programados bajo lenguaje Python.
- ResNet50 como arquitectura de red de deep learning.

5.8.5. Lineamientos de la categorización y procesamiento de imágenes

Se estableció la necesidad de generar un mínimo de 50 fotografías, a partir de la muestra de imágenes capturadas y procesadas (ver figura 12) en cada categoría, como resultado del proceso de captura de fotografías (en campo): se obtuvo una muestra de 50 fotografías por cada categoría, dando un total de 200 imágenes.

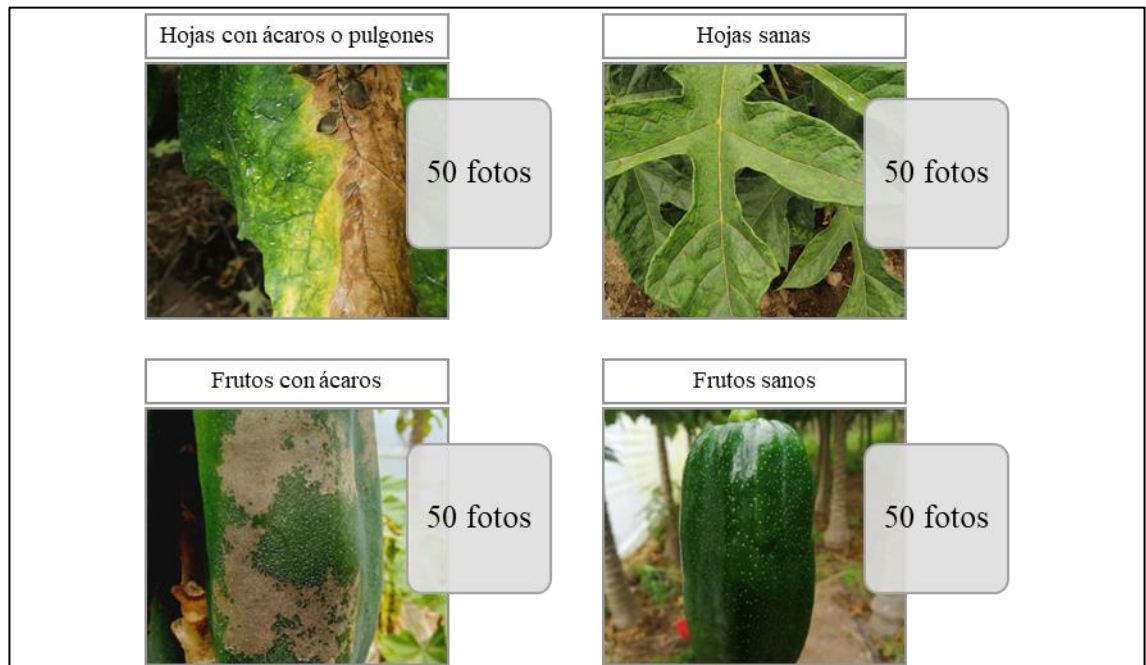


Figura 11. Tipos de plagas y cantidad de fotografías capturadas y procesadas

Cada fotografía se aumentó en 200 veces o epochs, dando un total de 30000 fotografías procesadas en el entrenamiento ($75 \text{ fotos} \times 4 \text{ categorías} \times 200 \text{ epochs} = 60000 \text{ fotografías}$).

5.9. Entrenamiento de la red neuronal y pruebas de clasificación

En esta fase con base en el estudio de Pereyra (2020), se empleó Google Colaboratory para el entrenamiento de la red neuronal. Este entorno permite importar Keras y una variedad de otras librerías (como opencv-python) útiles para el entrenamiento de la red neuronal, así como establecer un entorno de ejecución basado en GPU (requerido en el entrenamiento). En dicho entorno se implementó el código de entrenamiento en un archivo/interfaz denominado “notebook”, el cual ejecuta sus líneas de código secuencial o aleatoriamente. De manera general, el entrenamiento se efectuó mediante a los siguientes pasos:

5.9.1. Importación de paquetes y librerías

Destacan las librerías: keras (biblioteca de redes neuronales), opencv-python (tratamiento de imágenes), numpy (gestión de grandes vectores y matrices multidimensionales), matplotlib (generación de gráficos a partir de vectores y matrices), scipy (herramientas y algoritmos matemáticos), sklearn (biblioteca para aprendizaje automático en Python).

```
import os
import glob
import cv2
import math
import numpy as np
import random
import matplotlib.pyplot as plt
from matplotlib.pyplot import imshow
%matplotlib inline
from skimage import io
from keras import layers
from keras.layers import Input, Add, Dense, Activation, ZeroPadding2D, BatchNormalization,
from keras.models import Model, load_model
from keras.preprocessing import image
from keras.utils import layer_utils
from keras.utils.data_utils import get_file
from keras.applications.imagenet_utils import preprocess_input
from keras.utils.vis_utils import plot_model
from keras.initializers import glorot_uniform
import scipy.misc
import imageio
```

Figura 12. Importación de librerías, notebook de entrenamiento de la red neuronal

5.9.2. Cargar datos (imágenes)

Para cargar las fotografías al entorno de desarrollo, se vinculó o enlazó a Google drive:

```
from google.colab import drive
drive.mount('/content/drive')
```

Figura 13. Vinculación a Google drive, notebook de entrenamiento de la red neuronal

Posteriormente, se codificó la clase para cargar datos (imágenes y clasificación):

```
def load_dataset(directory, classes):
    """
    Cargar el dataset de imágenes y pre-procesarlas a un tamaño de 256 x 256 x 3.

    Argumentos:
    directory -- ruta del directorio/carpeta
    classes -- lista de categorías

    Retorna:
    X_orig -- np.array contenedor de todas las imágenes
    y_orig -- np.array contenedor de las etiquetas de todas las imágenes
    """
    paths = [] # almacena la ruta de las imágenes
    y_orig = [] # almacena el número de categoría
    for category in classes:
        flower_path = os.path.join(directory, category, '*')
        flower_paths = glob.glob(flower_path)
        for img_path in flower_paths:
            if img_path.endswith('.jpg'):
                paths.append(img_path)
                y_orig.append(classes.index(category))

    X_orig = []
    for i in range(len(paths)):
        if paths[i].endswith('.jpg'):
            img = cv2.imread(paths[i])
            img_res = cv2.resize(img, dsize=(256, 256), interpolation=cv2.INTER_AREA)
            X_orig.append(img_res)

    X_orig = np.array(X_orig)
    y_orig = np.array(y_orig)
    y_orig = y_orig.reshape((1, y_orig.shape[0]))

    return X_orig, y_orig, paths
```

Figura 14. Función `load_dataset`, notebook de entrenamiento de la red neuronal

Se cargaron los datos del directorio de Google drive en las matrices correspondientes:

```
#convertir etiquetas en matrices
def convert_to_one_hot(Y, C):
    Y = np.eye(C)[Y.reshape(-1)].T
    return Y

#directorio de almacenamiento de las fotografías
directory = "/content/drive/MyDrive/Colab Notebooks/plantasbabacocc"
#categorias
classes = ["frutos_acaro", "frutos_sano", "hojas_acapul", "hojas_sanas"] # 0 a 3

X_orig, Y_orig, paths = load_dataset(directory, classes)

# Normalizar vector de imágenes
X = X_orig/255.

# Convertir etiquetas en matrices
y = convert_to_one_hot(Y_orig, 4).T

print ("cantidad = " + str(X.shape[0]))
print ("X: " + str(X.shape))
print ("y: " + str(y.shape))
```

Figura 15. Inicialización de matrices y carga de datos, notebook de entrenamiento de la red neuronal

5.9.3. Programar y configurar sobre la arquitectura de aprendizaje profundo ResNet50

Se programó el “bloque de identidad”, el cual es un bloque estándar utilizado en ResNets, mismo que corresponde a la activación de entrada y la activación de salida con las mismas dimensiones (ver Figura 16).

A continuación, se programó el "bloque convolucional" de ResNet, el cual se emplea cuando no coinciden las dimensiones de entrada y salida. A diferencia del “bloque de identidad”, éste emplea una capa CONV2D en la ruta principal (ver Figura 17).

Finalmente se programó el proceso ResNet50 el cual, mediante un proceso interactivo, aplica procesos de relu (anula los valores negativos) y maxpool (encuentra el valor máximo de una ventana de muestra y lo pasa como resumen de dicha área. Reduciendo el tamaño de los datos) al dataset (ver Figura 18).

```
def identity_block(X, f, filters, stage, block, reg=1e-4):
    """
    Implementación del bloque de identidad (sobre 3 capas ocultas)

    Argumentos:
    X -- entrada del tensor de forma (m, n_H_prev, n_W_prev, n_C_prev)
    f -- entero, forma de la ventana convolucional de la ruta principal
    filters -- número de filtros en las capas convolucionales de la ruta principal
    stage -- entero, usado para nombrar las capas, dependiendo de su posición en la red
    block -- cadena / carácter, utilizado para nombrar las capas, dependiendo de su posición en la red

    Retorna:
    X -- salida del bloque de identidad, tensor de forma (n_H, n_W, n_C)
    """
    # Definiendo nombres base
    conv_name_base = 'res' + str(stage) + block + '_branch'
    bn_name_base = 'bn' + str(stage) + block + '_branch'

    # Recuperando filtros
    F1, F2, F3 = filters
```

Figura 16. Bloque de identidad, notebook de entrenamiento de la red neuronal

```
def convolutional_block(X, f, filters, stage, block, s = 2, reg=1e-4):
    """
    Implementación del bloque convolucional

    Argumentos:
    X -- entrada del tensor de forma (m, n_H_prev, n_W_prev, n_C_prev)
    f -- entero, forma de la ventana convolucional de la ruta principal
    filters -- número de filtros en las capas convolucionales de la ruta principal
    stage -- entero, usado para nombrar las capas, dependiendo de su posición en la red
    block -- cadena / carácter, utilizado para nombrar las capas, dependiendo de su posición en la red
    s -- Entero, especifica el paso que se utilizará

    Retorna:
    X -- salida del bloque convolucional, tensor de forma (n_H, n_W, n_C)
    """
    # definiendo nombres base
    conv_name_base = 'res' + str(stage) + block + '_branch'
    bn_name_base = 'bn' + str(stage) + block + '_branch'

    # Recuperando filtros
    F1, F2, F3 = filters

    # guardando el valor de entrada
    X_shortcut = X

    ##### Ruta principal #####
    # Primer componente
    X = Conv2D(F1, (1, 1), strides = (s,s), padding='valid',
               name = conv_name_base + '2a', kernel_initializer = glorot_uniform(seed=0),
               kernel_regularizer=l2(reg))(X)
    X = BatchNormalization(axis = 3, name = bn_name_base + '2a')(X)
    X = Activation('relu')(X)
```

Figura 17. Bloque convolucional, notebook de entrenamiento de la red neuronal

```
def ResNet50(input_shape = (256, 256, 3), classes = 4, reg=1e-4):
    """
    Implementación de RESNET50 con al arquitectura:
    CONV2D -> BATCHNORM -> RELU -> MAXPOOL -> CONVBLOCK -> IDBLOCK*2 -> CONVBLOCK -> IDBLOCK*3
    -> CONVBLOCK -> IDBLOCK*5 -> CONVBLOCK -> IDBLOCK*2 -> AVGPPOOL -> TOPLAYER

    Argumentos:
    input_shape -- formas de las imágenes del dataset
    classes -- entero, número de clases

    Retorna:
    model -- un Model() o modelo instanciado en Keras
    """
    # Defina la entrada como un tensor con forma input_shape
    X_input = Input(input_shape)

    # Relleno/padding cero
    X = ZeroPadding2D((3, 3))(X_input)

    # Paso 1
    X = Conv2D(128, (7, 7), strides = (2, 2), name = 'conv1',
               kernel_initializer = glorot_uniform(seed=0), kernel_regularizer=l2(reg))(X)
    X = BatchNormalization(axis = 3, name = 'bn_conv1')(X)
    X = Activation('relu')(X)
    X = MaxPooling2D((3, 3), strides=(2, 2))(X)
```

Figura 18. Proceso ResNet50, notebook de entrenamiento de la red neuronal

5.9.4. Configuración y ejecución del entrenamiento

Se configuró el aumento o argumentación de datos (imágenes), para evitar el sobreajuste de las imágenes, considerando únicamente el desplazamiento y rotación horizontal de la imagen.

```

gen = ImageDataGenerator(featurewise_center=False, # establece la media de entrada en 0 so
                          samplewise_center=False, # establece cada media de la muestra en
                          featurewise_std_normalization=False, # divide las entradas por st
                          samplewise_std_normalization=False, # divide cada entrada por su
                          zca_whitening=False, # aplica blanqueamiento ZCA
                          rotation_range=0, # rota las imágenes aleatoriamente en el rango
                          zoom_range = 0.1, # amplía la imagen aleatoriamente
                          width_shift_range=0.2, # desplaza las imágenes de forma aleatoria
                          height_shift_range=0.2, # desplaza las imágenes de forma aleatori
                          horizontal_flip=True, # voltea las imágenes al azar
                          vertical_flip=False) # voltea las imágenes al azar

gen.fit(X_train)

```

Figura 19. Configuración del proceso de argumentación, notebook de entrenamiento de la red neuronal

Se declaró la variable “model” para la asignación y configuración de ResNet50 sobre los siguientes parámetros:

- *Optimización Adam*: método de descenso de gradiente estocástico que se basa en la estimación adaptativa de momentos de primer y segundo orden.
- *Pérdidas Categorical_crossentropy*: función de pérdida para el modelo de clasificación de clases múltiples, el cual posee dos o más etiquetas de salida
- “Exactitud”: a fin de medir la calidad del modelo de machine learning en tareas de clasificación.

```

# Construye el gráfico del modelo
model = ResNet50(input_shape = (256, 256, 3), classes = 4)

# Configura el proceso de aprendizaje
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

#Reduce la tasa de aprendizaje cuando una métrica ha dejado de mejorar
reduce_lr = ReduceLROnPlateau(monitor='val_accuracy', factor=0.2, patience=5, min_lr=1e-4)

#entrenamiento y almacenamiento de detalles
cache = model.fit(gen.flow(X_train, Y_train, batch_size=40),
                  epochs = 200, validation_data = (X_test, Y_test),
                  steps_per_epoch = X_train.shape[0] // 40,
                  callbacks=[reduce_lr])

```

Figura 20. Configuración del entrenamiento, notebook de entrenamiento de la red neuronal

Finalmente, se ejecutó el proceso de entrenamiento de la red, tal como se observa en la siguiente figura 22. A medida que avanza el proceso, la métrica de exactitud varía con una tendencia global positiva.

```

Epoch 61/200
3/3 [=====] - 3s 797ms/step - loss: 2.0568 - accuracy: 0.9198 - val_loss: 6.4047 - val_accuracy: 0.1875
Epoch 62/200
3/3 [=====] - 2s 872ms/step - loss: 2.2533 - accuracy: 0.9167 - val_loss: 6.9195 - val_accuracy: 0.1875
Epoch 63/200
3/3 [=====] - 2s 579ms/step - loss: 1.9884 - accuracy: 0.9838 - val_loss: 6.9228 - val_accuracy: 0.1875
Epoch 64/200
3/3 [=====] - 3s 803ms/step - loss: 2.1344 - accuracy: 0.8865 - val_loss: 6.7232 - val_accuracy: 0.1875
Epoch 65/200
3/3 [=====] - 3s 777ms/step - loss: 1.9662 - accuracy: 0.9792 - val_loss: 6.4740 - val_accuracy: 0.1875
Epoch 66/200
3/3 [=====] - 3s 759ms/step - loss: 1.9897 - accuracy: 0.9688 - val_loss: 6.1915 - val_accuracy: 0.1875
Epoch 67/200
3/3 [=====] - 2s 626ms/step - loss: 2.1519 - accuracy: 0.9233 - val_loss: 5.5026 - val_accuracy: 0.2500
Epoch 68/200
3/3 [=====] - 2s 631ms/step - loss: 2.9235 - accuracy: 0.9304 - val_loss: 5.5698 - val_accuracy: 0.1875
Epoch 69/200
3/3 [=====] - 3s 759ms/step - loss: 1.9677 - accuracy: 0.9802 - val_loss: 5.7982 - val_accuracy: 0.1875

```

Figura 21. Entrenamiento de la red neuronal, notebook de entrenamiento de la red neuronal

Como primer resultado del entrenamiento, el modelo alcanzó una exactitud del 93,75% y como segundo resultado se obtuvo el 100% de exactitud para procesos de clasificación, para la obtención de estos resultados se hizo 2 entrenamientos mediante la red neuronal el cual consiste en introducir todas las imágenes existentes en el Google drive. Se llegó a la conclusión que en la red neuronal mientras más pruebas se vaya realizando, el entrenamiento se irá perfeccionando como se puede observar en las figuras 23 y 24.

```

# Evaluación del rendimiento del modelo
preds = model.evaluate(X_test, Y_test)
print ("Pérdida de la prueba = " + str(preds[0]))
print ("Exactitud de la prueba = " + str(preds[1]))

1/1 [=====] - 0s 78ms/step - loss: 2.0070 - accuracy: 0.9375
Pérdida de la prueba = 2.0069501399993896
Exactitud de la prueba = 0.9375

```

Figura 22. Resultado de la evaluación del primer rendimiento del modelo, notebook de entrenamiento de la red neuronal

```

# Evaluación del rendimiento del modelo
preds = model.evaluate(X_test, Y_test)
print ("Pérdida de la prueba = " + str(preds[0]))
print ("Exactitud de la prueba = " + str(preds[1]))

1/1 [=====] - 0s 102ms/step - loss: 1.7361 - accuracy: 1.0000
Pérdida de la prueba = 1.7360687255859375
Exactitud de la prueba = 1.0

```

Figura 23. Resultado de la evaluación del segundo rendimiento del modelo, notebook de entrenamiento de la red neuronal

Para reflejar la efectividad de los resultados del test de entrenamiento se tabula los datos del primer entrenamiento con las imágenes más relevantes entrenadas por la red neuronal mediante la comparación con un experto, como se muestra talen la tabla 9.

#	EXPERTO	AFECCION	PREDICCIÓN	RED NEURONAL	RESULTADO COMPARATIVO (EX y RED)
1	frutos_sano	sano	sano	frutos_sano	SI
4	frutos_acaro	plaga	sano	frutos_acaro	SI
5	hojas_acapul	plaga	sano	hojas_acapul	SI
6	hojas_sana	sano	sano	hojas_sana	SI
9	frutos_sano	sano	sano	frutos_sano	SI
11	frutos_sano	sano	sano	frutos_sano	SI
15	hojas_acapul	plaga	acaro	hojas_acapul	SI
16	hojas_sana	sano	acaro	hojas_acapul	NO
19	frutos_sano	sano	sano	frutos_sano	SI
20	frutos_sano	sano	sano	frutos_sano	SI
21	hojas_acapul	plaga	sano	hojas_acapul	SI
22	hojas_sana	sano	sano	hojas_sana	SI
24	frutos_acaro	plaga	acaro	frutos_acaro	SI
27	frutos_acaro	plaga	sano	frutos_sano	NO
28	hojas_acapul	plaga	sano	hojas_acapul	SI
29	hojas_sana	sano	acaro	hojas_sana	SI
30	frutos_acaro	plaga	sano	frutos_acaro	SI
31	frutos_acaro	plaga	acaro	frutos_acaro	SI
33	frutos_sano	sano	sano	frutos_sano	SI
37	hojas_acapul	plaga	sano	hojas_acapul	SI
38	hojas_sana	sano	acaro	hojas_sana	SI
43	hojas_acapul	plaga	sano	hojas_acapul	SI
45	hojas_sana	sano	acaro	hojas_sana	SI
47	hojas_acapul	plaga	sano	hojas_acapul	SI
49	hojas_sana	sano	sano	hojas_sana	SI
52	frutos_acaro	plaga	acaro	frutos_acaro	SI
53	frutos_acaro	plaga	sano	frutos_acaro	SI
54	hojas_acapul	plaga	sano	hojas_acapul	SI
57	hojas_sana	sano	sano	hojas_sana	SI
58	frutos_sano	sano	sano	frutos_sano	SI
62	frutos_acaro	plaga	sano	frutos_acaro	SI
63	frutos_acaro	plaga	sano	frutos_acaro	SI
66	hojas_acapul	plaga	sano	hojas_acapul	SI
67	hojas_sana	sano	sano	hojas_sana	SI
69	frutos_acaro	plaga	sano	frutos_acaro	SI
70	frutos_acaro	plaga	acaro	frutos_acaro	SI
71	frutos_sano	sano	sano	frutos_sano	SI
72	hojas_acapul	plaga	acaro	hojas_acapul	SI
73	hojas_sana	sano	sano	hojas_sana	NO
74	frutos_acaro	plaga	acaro	frutos_acaro	SI

75	hojas_acapul	plaga	acaró	hojas_acapul	SI
78	hojas_sana	sano	sano	hojas_sana	SI
79	frutos_acaro	plaga	sano	frutos_acaro	SI
84	frutos_acaro	plaga	sano	frutos_acaro	SI
85	hojas_acapul	plaga	sano	hojas_acapul	SI
89	hojas_sana	sano	sano	hojas_sana	SI
90	frutos_sano	sano	sano	frutos_sano	SI
91	frutos_sano	sano	sano	frutos_sano	SI
96	hojas_acapul	plaga	sano	hojas_acapul	SI
98	hojas_sana	sano	acaró	hojas_sana	SI
				Aciertos	37
				Errores	3
Predicciones		Frecuencia	Porcentaje		
Aciertos		47	93,75%		
Errores		3	6,25%		
Total		50	100%		

Tabla 9. Tabulación entre resultados de experto y el entrenamiento de la red neuronal



Figura 24. Cuadro estadístico de la tabulación

En la figura 26 se puede observar que el resultado reflejado por medio del entrenamiento a través de la red neuronal fue efectivo ya que detecta el tipo de plaga (pulgonés) que se presenta en la hoja de babaco.

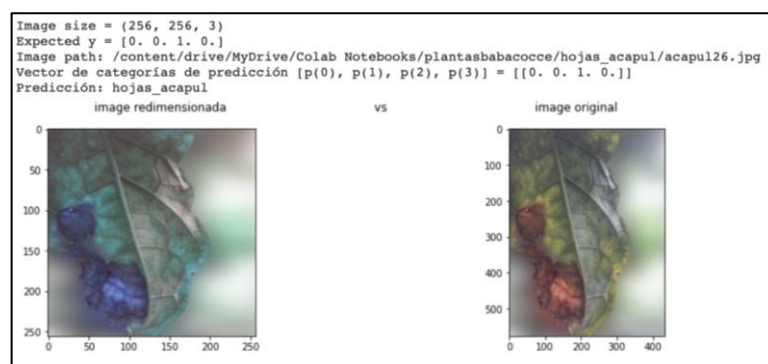


Figura 25. Resultado de la efectividad alcanzada del entrenamiento (pulgonés)

A su vez en la figura 27 se puede observar que el resultado reflejado por medio del entrenamiento a través de la red neuronal fue efectivo ya que detecta el tipo de plaga (ácaros) que se presenta en el fruto del babaco.

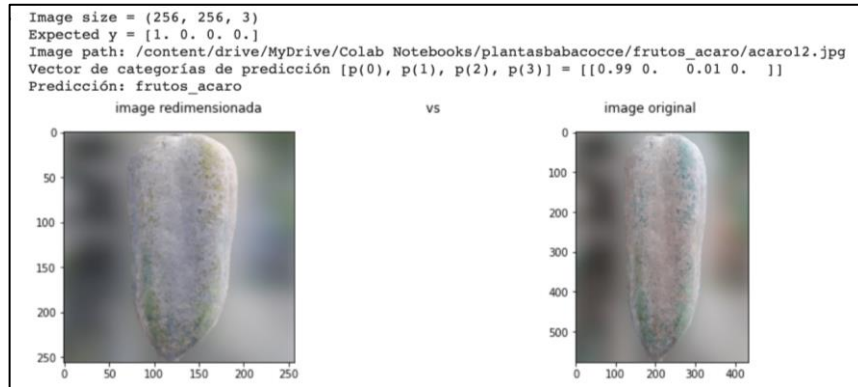


Figura 26. Resultado de la efectividad alcanzada del entrenamiento (pulgones)

A su vez en la figura 28 se puede observar que el resultado reflejado por medio del entrenamiento a través de la red neuronal fue efectivo ya que detecta que la imagen procesada en el entrenamiento pertenece a un fruto sano sin afectación por plagas.

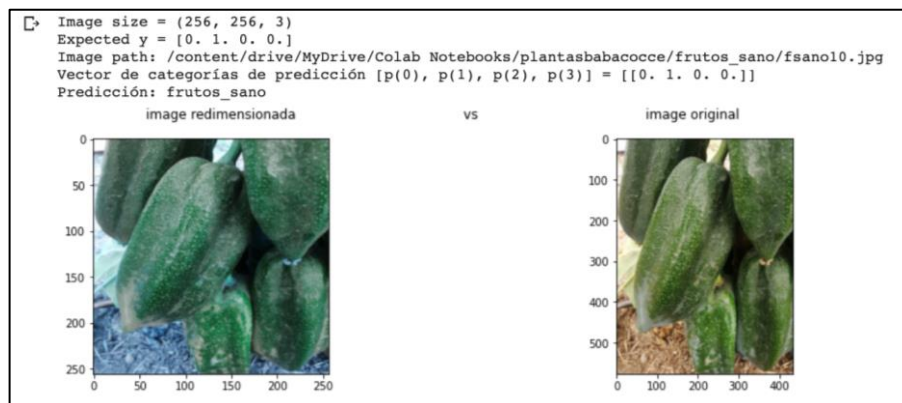


Figura 27. Predicciones del modelo, notebook de entrenamiento de la red neuronal

Finalmente, se procedió a guardar el modelo generado, para efectos del desarrollo del siguiente objetivo de investigación.

```

# Guardar el Modelo
model.save('/content/drive/MyDrive/Colab Notebooks/model1.h5')
  
```

Figura 28. Modelo almacenado, notebook de entrenamiento de la red neuronal

5.9.5. Resultados del Testing o prueba del modelo

Una vez efectuado el entrenamiento de la red neuronal, se procede a su validación, es decir se presentan diferentes datos de las cuatro categorías, donde se pudo observar errores muy bajos en la categorización de las imágenes.

Para visualizar la validación de la red se dispone de imágenes similares a las ya efectuadas durante los dos entrenamientos, como consecuencia, se procedió con la segmentación del primer modelo para su reentrenamiento a fin de descartar problemas a nivel de los datasets.

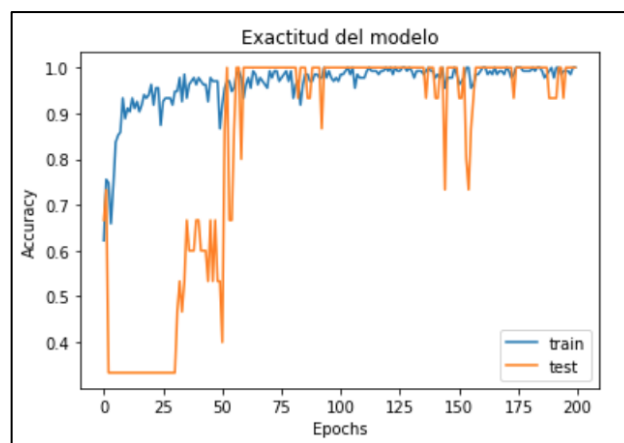


Figura 29. Resultado de la exactitud

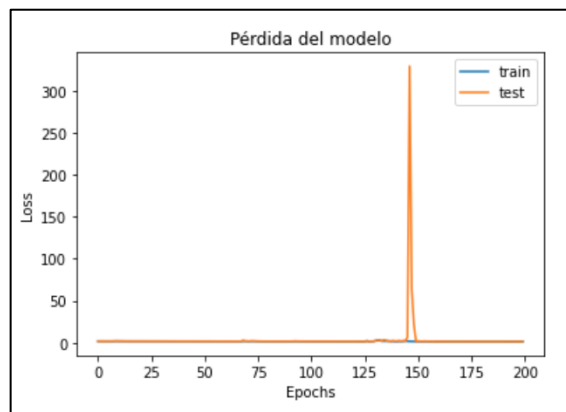


Figura 30. Resultado de la pérdida del modelo

Segmentación del segundo modelo para su reentrenamiento, a fin de descartar problemas a nivel de los datasets.

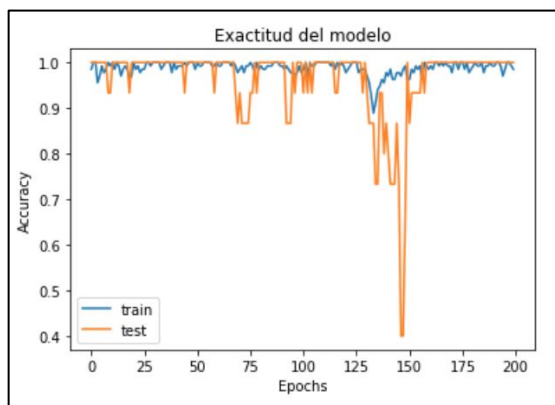


Figura 31. Resultado de la exactitud

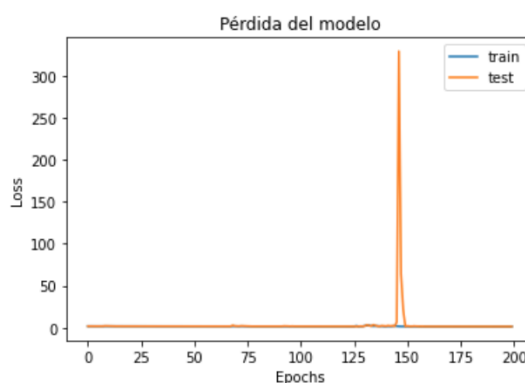


Figura 32. Resultado de la pérdida del modelo

5.9.5.1. Resultado del test de los entrenamientos

Número de entrenamiento	Pérdidas
1	1.6655
2	1.2720

Tabla 10. Resultado de los dos entrenamientos

5.9.5.2. Estadística de predicciones

El nivel de exactitud del modelo es el máximo posible (100%), sin embargo, a medida que se realizan los entrenamientos, las pérdidas pueden ir disminuyendo.

En la aplicación del modelo generado a un dataset de imágenes de 50 elementos (25 con plaga tipo ácaro y 25 sin plaga), se obtuvieron los resultados plasmados en la siguiente tabla: 78% de acierto general y 22% de error.

Predicciones	Frecuencia	Porcentaje
Aciertos	39	78%
Errores	11	22%
Total	50	100%

Tabla 11. Resultado de las predicciones de la estadística

5.9.6. Historias de usuario

Historia de usuario: donde el usuario requiere que la aplicación móvil permita seleccionar la carpeta almacenada dentro del Google drive.

HISTORIA DE USUARIO (HU)			
Código HU:	HU001	Fecha:	12/04/2021
Sprint:	1	Prioridad:	Alta
Actores:	Usuario	Puntos:	3
Descripción: Como usuario requiero que la aplicación móvil permita seleccionar la carpeta almacenada dentro del Google drive.			
Detalles de la HU: <ul style="list-style-type: none"> El usuario ingresa a la aplicación El usuario da clic en el botón seleccionar imagen. El aplicativo muestra la lista de las carpetas almacenadas. 			
Restricciones: Solo el usuario que tenga la aplicación puede realizar el proceso.			
Criterios de aceptación: <ul style="list-style-type: none"> La aplicación debe mostrar las carpetas del Google drive. 			
DoD: Se realizó la prueba funcional de esta historia de usuario dando como resultado un estatus exitoso.			

Tabla 12. Historia de usuario 1. Selección de la carpeta dentro de Google drive

Historia de usuario: donde el usuario requiere que la aplicación móvil permita seleccionar una imagen dentro de las carpetas del Google drive.

HISTORIA DE USUARIO (HU)			
Código HU:	HU002	Fecha:	12/04/2021
Sprint:	2	Prioridad:	Alta
Actores:	Usuario	Puntos:	3
Descripción: Como usuario requiero que la aplicación móvil permita seleccionar una imagen dentro de las carpetas del Google drive.			
Detalles de la HU: <ul style="list-style-type: none"> • El usuario ingresa a la aplicación • El usuario elige una imagen dentro de las carpetas del Google drive. • El aplicativo muestra la interfaz con la imagen seleccionada. 			
Restricciones: Solo el usuario que tenga la aplicación puede realizar el proceso			
Criterios de aceptación: <ul style="list-style-type: none"> • La aplicación debe mostrar la imagen seleccionada en la interfaz. 			
DoD: Se realizó la prueba funcional de esta historia de usuario dando como resultado un estatus exitoso.			

Tabla 13. Historia de usuario 2. Selección de imágenes de las carpetas almacenadas en el drive

Historia de usuario: donde el usuario requiere que la aplicación móvil detecte el tipo de plaga de la imagen seleccionada.

HISTORIA DE USUARIO (HU)			
Código HU:	HU003	Fecha:	12/04/2021
Sprint:	3	Prioridad:	Alta
Actores:	Usuario	Puntos:	3
Descripción: Como usuario requiero que la aplicación móvil detecte el tipo de plaga de la imagen seleccionada.			
Detalles de la HU: <ul style="list-style-type: none"> • El usuario ingresa a la aplicación. • El aplicativo muestra el botón obtener predicción. • El usuario da clic en obtener predicción de la imagen seleccionada. 			

<ul style="list-style-type: none"> El aplicativo muestra el resultado de la predicción.
Restricciones: Solo el usuario que tenga la aplicación puede realizar el proceso
Criterios de aceptación: <ul style="list-style-type: none"> La aplicación debe mostrar el resultado de la predicción del tipo de plaga.
DoD: Se realizó la prueba funcional de esta historia de usuario dando como resultado un estatus exitoso.

Tabla 14. Historia de usuario 3. Predicción del tipo de plaga

5.10. Resultado de la validación de la metodología de desarrollo evolutivo

En esta fase de validación se establece Android studio para el desarrollo de la interfaz donde se puede observar el ícono de la aplicación móvil.



Figura 33. Ícono de la aplicación

En la figura 34 se observa la interfaz de inicio de la aplicación, en la cual nos permite seleccionar la imagen de clasificación del entrenamiento para ver la información del tipo de plaga.



Figura 34. Interfaz para la selección de imágenes

En la figura 35 se puede observar la interfaz que permite seleccionar la carpeta donde se encuentran almacenadas las fotos entrenadas por la red neuronal.

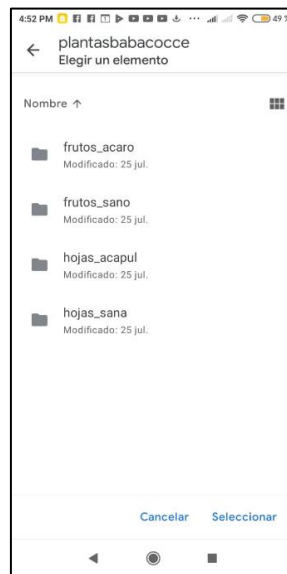


Figura 35. Interfaz de las carpetas que contiene las imágenes procesadas

En la figura 36 se puede observar las imágenes almacenadas dentro de las carpetas ya categorizadas.

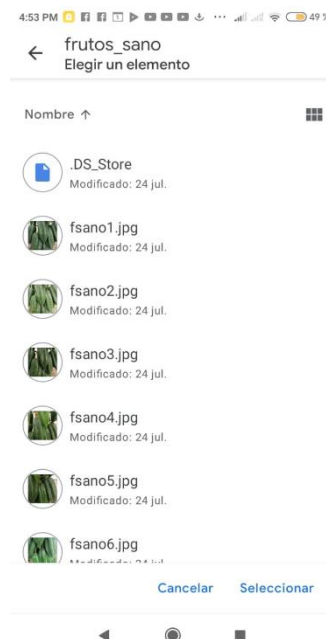


Figura 36. Interfaz de las imágenes almacenadas

En la figura 37 se puede observar la imagen seleccionada para su respectivo análisis.



Figura 37. Interfaz que muestra la imagen seleccionada

En la figura 38 se puede observar el resultado de la predicción mostrando si las hojas o el fruto presentan algún tipo de plaga o si se encuentra sana.



Figura 38. Interfaz donde muestra los resultados

5.10.1. Plan de validación de la aplicación (ejecución de pruebas)

Ejecución de pruebas: Determinación de la validez de la historia de usuario número

1.

Historia de usuario: HU001		Autor del Caso de Prueba: Isaac López	
ID/Nombre Caso de Prueba: CP001		Fecha de Creación: 20 de Junio del 2021	
Versión 1.0		Fecha de Ejecución: 08 de Julio del 2021	
Flujo de pasos de la Prueba:			
Nro.	Descripción del paso	Resultado Esperado	Resultado Obtenido
1	La aplicación le permite el ingreso al usuario.	La aplicación móvil le permite el ingreso al usuario.	La aplicación si le permite ingresar al usuario.
2	El usuario selecciona la imagen.	La aplicación móvil permite seleccionar la imagen.	La aplicación si le permite al usuario seleccionar la imagen.
3	El aplicativo muestra la lista de las carpetas almacenadas.	La aplicación móvil muestra la lista de las carpetas.	La aplicación móvil si muestra la lista de las carpetas.
Decisión de Aprobación del Caso de Prueba: Aprobó: _x_ Fallo: ____			
Nombre y firma del Probador		Isaac López	
Nombre y firma del Cliente			
Fecha de Aprobación del Caso de Prueba: 10 de Julio del 2021			

Tabla 15. Ejecución de pruebas de la historia de usuario número 1.

Ejecución de pruebas: Determinación de la validez de la historia de usuario número

2.

Historia de usuario: HU002		Autor del Caso de Prueba: Isaac López	
ID/Nombre Caso de Prueba: CP002		Fecha de Creación: 25 de Junio del 2021	
Versión 1.0		Fecha de Ejecución: 12 de Julio del 2021	
Flujo de pasos de la Prueba:			
Nro.	Descripción del paso	Resultado Esperado	Resultado Obtenido

1	El usuario elige una imagen dentro de las carpetas del Google drive.	La aplicación móvil permite elegir al usuario una imagen que esta almacenada dentro del drive.	La aplicación si le permite al usuario elegir una imagen que esta almacenada dentro del drive.
2	El aplicativo muestra la interfaz con la imagen seleccionada.	La aplicación móvil muestra la imagen seleccionada.	La aplicación si muestra la imagen seleccionada.
Decisión de Aprobación del Caso de Prueba: Aprobó: <u> x </u> Fallo: <u> </u>			
Nombre y firma del Probador		Isaac López	
Nombre y firma del Cliente			
Fecha de Aprobación del Caso de Prueba: 14 de Julio del 2021			

Tabla 16. Ejecución de pruebas de la historia de usuario número 2.

Ejecución de pruebas: Determinación de la validez de la historia de usuario número 3.

Historia de usuario: HU003		Autor del Caso de Prueba: Isaac López	
ID/Nombre Caso de Prueba: CP003		Fecha de Creación: 16 de Julio del 2021	
Versión 1.0		Fecha de Ejecución: 27 de Julio del 2021	
Flujo de pasos de la Prueba:			
Nro.	Descripción del paso	Resultado Esperado	Resultado Obtenido
1	El aplicativo muestra el botón obtener predicción.	El aplicativo móvil muestra el botón obtener predicción.	El aplicativo móvil si muestra el botón obtener predicción.
2	El usuario da clic en obtener predicción de la imagen seleccionada.	La aplicación móvil muestra la predicción de la imagen seleccionada.	La aplicación móvil si muestra la predicción de la imagen seleccionada.
Decisión de Aprobación del Caso de Prueba: Aprobó: _x_ Fallo: ____			
Nombre y firma del Probador		Isaac López	
Nombre y firma del Cliente			
Fecha de Aprobación del Caso de Prueba: 30 de Julio del 2021			

Tabla 17. Ejecución de pruebas de la historia de usuario número 3.

5.11. Validación de la hipótesis

5.11.1. Juicio de experto

Para la validación de la hipótesis se aplicó el juicio de expertos, Master en diseño y animación web Gonzalo Ramon Balverde Escobar, además del señor Héctor Mecías agricultor dedicado a plantaciones de babaco y del Ingeniero Agrónomo Miguel Ángel Beltrán Sánchez supieron ver y constatar por medio de un cuestionario que el entrenamiento de la red neuronal alcanzó una exactitud considerable al proyecto de estudio para los procesos de clasificación de las imágenes acorde al ataque de las plagas por ácaros en los frutos y en las plagas de pulgones. Visualmente tiene diferentes tipos de manifestaciones que para el ojo del ser humano son muy sencillas de reconocer pero para la ejecución de pruebas se debe considerar un solo defecto tal como lo es en los ácaros, por lo cual supieron manifestar que la investigación desarrollada por parte del equipo de trabajo servirá de gran ayuda en la detección de plagas y a su vez servirá como motivo de estudio para que nuevos investigadores continúen con la fase de desarrollo para llevar un mejor y adecuado control sobre las mismas, esto se encuentra solventado en el documento que se muestra en el anexo 8.4.

6. CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones

- Para la elaboración del proyecto de investigación se estableció un esquema y por medio de investigaciones bibliográficas empleadas en proyectos de visión artificial se pudo establecer el modelo adecuado a entrenar de la red neuronal siendo Google Colab como plataforma de desarrollo, OpenCv para el entrenamiento de las imágenes y ResNet50 como arquitectura de la red Deep learning.
- Se obtuvo como resultado del primer entrenamiento un 93.75%, mientras que el segundo entrenamiento de la red neuronal se alcanzó el 100% de efectividad, lo cual permitió clasificar las imágenes para determinar el tipo de plaga.
- Para reflejar el resultado de la investigación realizada se complementa con una aplicación móvil.

6.2. Recomendaciones

- ❖ Desarrollar a futuro mediante otro trabajo de investigación los diferentes aspectos que conlleve a la detección de las plagas en plantaciones de babaco.
- ❖ Mediante la ayuda de ingenieros agrónomos complementar a futuro un adecuado control que permita diagnosticar y dosificar el tipo de plaga dentro de la aplicación móvil.
- ❖ Con el fin de dar una mejora en la aplicación se recomienda trabajar en tiempo real, lo cual ayude a futuro visualizar los resultados de una manera adecuada.

7. BIBLIOGRAFÍA

- [1] FAO, «Glosario de términos fitosanitarios,» 1995. [En línea]. Available: <http://www.fao.org/3/w3587e/w3587e03.htm>.
- [2] I. Zepeda, «Manejo sustentable de plagas agrícolas en México,» *Agricultura, sociedad y desarrollo*, vol. 15, nº 1, pp. 99-108, 2018.
- [3] Food and Agriculture Organization of the United Nations, «El estado mundial de la agricultura y la alimentación,» 2014. [En línea]. Available: <http://www.fao.org/3/a-i4040s.pdf>.
- [4] F. Benito, «El calentamiento dispara las plagas de los cultivos,» 2020. [En línea]. Available: <https://verdeyazul.diarioinformacion.com/calentamiento-plagas-cultivos.html>.
- [5] K. Lucero, «Manabí, la tierra más fértil del país donde la pobreza aún galopa,» 2020. [En línea]. Available: <https://www.revistagestion.ec/economia-y-finanzas-analisis/manabi-la-tierra-mas-fertil-del-pais-donde-la-pobreza-aun-galopa>.
- [6] El Comercio, «Las plagas impactan en seis cultivos principales,» 2019. [En línea]. Available: <https://www.pressreader.com/ecuador/el-comercio-ecuador/20190916/281492163021720>.
- [7] M. Villarreal, E. Villa, L. Cira, M. Estrada, F. Parra y S. Santos, «El género *Bacillus* como agente de control biológico y sus implicaciones en la bioseguridad agrícola,» *Revista mexicana de fitopatología*, vol. 36, nº 1, pp. 95-130, 2018.
- [8] N. Rosero, J. Cabrera, M. Anrango, M. Yandún y S. Lascano, «Detección de enfermedades en cultivos de papa usando procesamiento de imágenes,» *Revista CUMBRES*, vol. 6, nº 1, pp. 43-52, 2020.
- [9] C. Sullca, C. Molina, C. Rodríguez y T. Fernández, «Detección de enfermedades y plagas en las hojas de arándanos utilizando técnicas de visión

- artificial,» *PERSPECTIV@S. Revista de Tecnología e Información*, vol. 15, nº 15, pp. 32-39, 2018.
- [10 J. Piscoya, Sistema de Visión Artificial para apoyar en la identificación de plagas y enfermedades del Cultivo de Sandía en el Distrito de Ferreñafe, Chiclayo: Universidad Católica Santo Toribio De Mogrovejo, 2019.
- [11 A. Proaño, Evaluación de diferentes tipos de embalaje e índices de madurez en Babaco, Cevallos: Universidad Técnica De Ambato, 2020.
- [12 INIAP, «Babaco, fruto con potencial en el Ecuador y el mundo,» *Revista informativa del Instituto Nacional Autónomo de Investigaciones Agropecuarias*, pp. 35-43, 1997.
- [13 V. Maeda, «Comparación de arquitecturas de redes neuronales convolucionales para la clasificación de enfermedades en tomate,» Universidad Autónoma de Zacatecas, Zacatecas, 2019.
- [14 E. Huerta, V. González, H. Rodríguez y L. Amabilis, «Detección de enfermedades foliares con arquitecturas de redes neuronales convolucionales,» *RINDERESU*, vol. 5, nº 1, pp. 18-40, 2020.
- [15 A. d. A. I. d. Cañar(AAIC), El cultivo de babaco en invernadero (Carica pentágona), Quito: Editorail Abya Yala, 2003.
- [16 D. J. Matich, «Redes Neuronales: Conceptos básicos y Aplicaciones.,» Rosario, 2001.
- [17 G. Ana, M. Francisco y P. Alpha, «Técnicas y Algoritmos básicos de visión artificial,» Universidad de la Rioja, 2006.
- [18 A. F. Montoro, Phyton tres al descubierto, Mexico: Alfaomega grupo editor, 2013.
- [19 T. GIS, «CursosGIS,» Grupo TYC GIS, [En línea]. Available: <https://www.cursosgis.com/que-es-google-colab/>.

- [20 I. A. c. T. p. d. comportamientos, «Daniel Conde Ortiz,» Universidad de
] Sevilla, Sevilla, 2018.
- [21 H. Rodríguez, «crehana,» [En línea]. Available:
] <https://www.crehana.com/ec/blog/desarrollo-web/que-es-opencv/>.
- [22 B. D. S. Differently, «piperlab,» 2020. [En línea]. Available:
] <https://piperlab.es/glosario-de-big-data/keras/>.
- [23 E. B. Herreras, «SPSS: UN INSTRUMENTO DE ANÁLISIS DE DATOS
] CUANTITATIVOS,» Universidad de León, España, 2005.
- [24 M. Rivera, «La red Residual,» agosto 2019. [En línea]. Available:
] http://personal.cimat.mx:8181/~mriviera/cursos/aprendizaje_profundo/resnet/resnet.html.
- [25 N. G. L. D. F. C. S. Pablo Thomas, «Ingeniería de Software en el desarrollo
] de Aplicaciones para Dispositivos,» *handle*, vol. 1, n° 1, p. 586, 2013.
- [26 R. Casado-Vara, *Introduccion a Html*, Salamaca: Ediones Ciudad de
] Salamanca, 2019.
- [27 Á. M. V. APONTE, *GUÍA COMPARATIVA DE FRAMEWORKS PARA LOS
] LENGUAJES HTML 5*, Pereira, 2014.
- [28 J. E. Pérez, *Introduccion a java script*, Libros web, 2019.
]
- [29 D. Martinez, *PHP Basico y Practico*, Medellin: ITM, 2012.
]
- [30 D. J. O. Ortega, “*Bootstrap y Laravel, herramientas para el desarrollo de
] aplicaciones web*”, Mazatlán, Sinaloa, 2019.
- [31 Á. B. J. C. C. M. G. F. H. P. J. R. d. L. D. S. S. P. T. Á. Z. Manuel Báez,
] *Introducción a Android*, Madrid: E.M.E, 2019.

- [32 M. G. G. Ó. P. M. PLuis Alberto Casillas Santillán, *Bases de datos en MySQL*,
] Barcelona, 2014.
- [33 S. E. M. B. P. P. Ariel Pasini, *Q-Scrum: una fusión de Scrum y el estándar*
] *ISO/IEC*, Buenos Aires, 2013.
- [34 L. T. Francisco Rueda, *Comparativa Kotlin y Java en desarrollo Android*,
] Malaga, 2019.
- [35 R. A. G. Orellana, *IMPLEMENTACIÓN DE UNA API DE INFORMACIÓN*
] *BANCARIA PARA LA SBIF*, Santiago de Chile, 2020.
- [36 E. Bahit, *Scrum y Xtreme Programming para programadores*, Buenos Aires:
] Creative, 2012.
- [37 «Aprende Web.Net,» Viernes julio 2018. [En línea]. Available:
] <http://aprende-web.net/librerias/>. [Último acceso: Viernes Mayo 2021].
- [38 «HostGator,» jueves Enero 2021. [En línea]. Available:
] <https://www.hostgator.mx/blog/todo-lo-que-necesitas-saber-sobre-los-dominios/>. [Último acceso: Viernes Mayo 2021].
- [39 R. L. Y. DEYNIZ, SISTEMA WEB BASADO EN LA GESTIÓN DE
] INCIDENCIASPARA MEJORAR EL SOPORTE INFORMÁTICO EN LA,
1 ed., vol. 1, Chimbote, Nueci, 2017, pp. 23-24.
- [40 M. Marquez, *Base de Datos, Costelo de la plana: Sapientia*, 2011.
]
- [41 C. D. R. SANLY APONTE GOMEZ, *Proyecto de Grado Tekhlar II como*
] *requisito para optar al título de Ingeniero en*, Bogota, 2011.
- [42 A. L. M. y. A. M. Adelaida Delgado, *Trello Aprendizaje para la enseñanza*, 1
] ed., vol. 1, Oviedo: Taller XX JENUI, 2014, p. 6.
- [43 M. Carlos y V. Jorge, *El babaco*, IPA, La Platina, 1985.
]

- [44 J. Quiza, «medium,» ciencia y datos, 14 mayo 2018. [En línea]. Available:
] <https://medium.com/datos-y-ciencia/modelos-cnn-en-la-clasificaci%C3%B3n-de-im%C3%A1genes-cl%C3%A1sicas-y-modernas-d072a6718689>.

8. ANEXOS

8.1. Anexo: Hoja de vida de investigadores



PATRICIO ALEJANDRO GAMBOA MURILLO

✉ patricio.gamboa3869@uto.edu.ec

☎ 0980267022

📍 Ambato - Ecuador

26 años

Soltero

OBJETIVO PERSONAL

Desempeñarme profesionalmente en el área de desarrollo de plataformas web y móviles.

Obtener más conocimientos en cualquier área de trabajo que me permitan crecer como persona y como profesional.

PROGRAMAS

MICROSOFT	★ ★ ★ ★ ★
JAVA	★ ★ ★ ★ ★
C++	★ ★ ★ ★ ★
EXCEL	★ ★ ★ ★ ★
KOTLIN	★ ★ ★ ★ ★
PHP	★ ★ ★ ★ ★

REFERENCIAS

- ✓ Español
- ✓ Inglés (Suficiencia)

EXPERIENCIA PROFESIONAL

De 22/11/2020 hasta la actualidad. **UNIVERSIDAD TÉCNICA DE COTOPAXI**
Tesis
Desarrollo de un prototipo para el reconocimiento de plagas en plantaciones de babaco bajo invernadero utilizando redes neuronales.

De 04/10/2020 a 25/02/2021 (Salcedo). **INLADEC S.A. (LACTEOS).**
Pasante
Desarrollo de un sistema de inventario.
Tareas realizadas:
Elaboración de una plataforma web que permita el ingreso de datos de materia prima, mediante la implementación de un sistema para el control de inventario.
Entrega del software, manual de usuario y manual técnico.

De 28/10/2020 a 15/02/2021 (Ciudad-País). **DESARROLLO DE UNA APLICACIÓN MOVIL PARA EL RESTAURANTE LA VAQUITA**
Materia Aplicaciones Móviles
Tareas realizadas:
Desarrollo e implementación de una App que permita la toma de pedidos.

De 10/03/2012 a 28/11/2012 (Ambato-Ecuador). **ANDYFER,**
Despachador
Tareas realizadas:
Atender y despachar los productos de la ferretería.

De 02/12/2012 a 07/2014 (Ambato-Ecuador). **ELECTROCONSTRUCCIONES,**
Ayudante de electricidad
Tareas realizadas:
Instalaciones domiciliarias. 110v

EDUCACIÓN

2021 **ESTUDIANTE DE INGENIERÍA EN INFORMÁTICA Y SISTEMAS COMPUTACIONALES (10mo)**
Latacunga - Ecuador Universidad Técnica de Cotopaxi

2012 **BACHILLER AGRÓNOMO**
Ambato - Ecuador Instituto Tecnológico Agropecuario "Luis A. Martínez".

HOJA DE VIDA

Isaac Samuel López Miniguano



1.- DATOS PERSONALES

PROVINCIA: Tungurahua

CIUDAD: Ambato

DIRECCIÓN: Pinllo

CELULAR: 0960155795 0979946076

CORREO ELECTRÓNICO: isaac.mukito@gmail.com

Nº CARNE CONADIS: No

TIPO DE SANGRE: O+

CEDULA: 1850590819

2.- INSTRUCCIÓN

Nivel de Instrucción	Nombre de la Institución	Especialización	Título	Registro SENESCYT
Primaria	José Joaquín Olmedo	Educación Básica	-	-
Secundaria	Instituto tecnológico "Rumiñahui"	Técnico de servicios aplicaciones informáticas	-	ME-REF.699869 27-08-2014
(Tercer nivel)	Universidad Técnica de Cotopaxi	ingeniería en informática y sistemas computacionales	Cursando último semestre (decimo)	-
Licencia profesional (tipo C)	Escuela de capacitación para conductores profesionales.	-	Conductor profesional	-

8.2. Anexo: Formulario de Entrevista



UNIVERSIDAD TÉCNICA DE COTOPAXI

FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS

CARRERA DE INGENIERÍA EN INFORMÁTICA Y SISTEMAS

COMPUTACIONALES

ENTREVISTA

¿Se considera al babaco como una fruta muy apetecida?

Si, ya que el babaco es una fruta que aporta nutrientes que son beneficiosos para mantener una buena salud y puede ser consumido en dulces, mermeladas, jugos, pulpas o en frescos.

¿Alrededor de cuantas plantaciones de babaco considera que existe en el Ecuador?

En el Ecuador hay aproximadamente más de 180 hectáreas sembradas con dicha fruta ya sea bajo invernadero o a cielo abierto, siendo la Provincia de Tungurahua quien ocupa el primer lugar con un 60% aproximadamente de la producción a nivel nacional.

¿Cuál es la altitud, temperatura y humedad requerida para la siembra de esta fruta?

Las plantas de babaco se adaptan muy fácilmente a los suelos situados entre los 1600 y 3200 metros de altitud, la temperatura adecuada para su producción está entre los 14 y 27°C con una humedad que va en el rango del 70 al 80% el mismo que es controlado dentro de los invernaderos.

¿Existen daños por plagas en las plantaciones de babaco que afectan a la cosecha?

Si existen tres tipos de plagas que dañan a la plantación de babaco los cuales son nemátodos, ácaros y pulgón, donde los nemátodos se encargan de atacar a las raíces del babaco interrumpiendo el paso de los nutrientes causando así el retraso del crecimiento de la planta, los ácaros atacan al fruto tornándolo de color castaño hasta que la fruta se pudre y al follaje donde forman colonias en el envés de las hojas y forman telarañas dañando así las hojas y el pulgón que se ubican en los brotes tiernos y en el envés de las hojas, succionan la sabia y transmiten virus.

¿Alguna vez a usado herramientas tecnológicas que permita detectar plagas en las plantaciones?

No, no he usado medios de tecnología para la detección de plagas

¿Considera usted que un prototipo puede ayudar en la fácil detección de plagas?

Si yo creo que la ayuda de la tecnología si nos puede ayudar a los agricultores no solo de babaco si no en general a prevenir y mejorar nuestros procesos de cultivo.

8.2.1. Anexo: Resultado de la entrevista

La entrevista que se le realizó al agricultor Héctor Mecías permitió recabar información fundamental y conocer el punto de vista donde manifestó sus criterios en relación a las preguntas desarrolladas considerando que el proyecto de investigación permita de una u otra manera ayudar en el reconocimiento de las plagas y así evitar más pérdidas económicas y de tiempo.

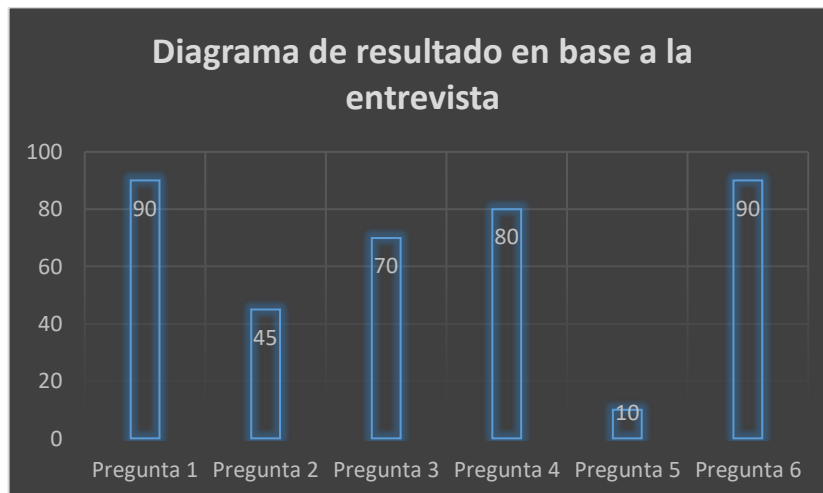


Figura 10. Diagrama de resultado de la entrevista

8.3. Anexo: Plagas que se presentan en las plantaciones de babaco.

PLAGAS QUE AFECTAN A LAS PLANTACIONES DE BABACO


Los cultivos de babaco bajo invernadero ocupan una gran extensión de hectáreas, pues se siembra desde pequeñas áreas hasta grandes áreas, ya sea, para consumo familiar o para comercio exterior. Actualmente en el Ecuador más específicamente en la provincia de Tungurahua es común encontrar cultivos de babaco bajo invernadero los cuales presentan problemas de ataque de plagas, que afectan en la producción de la planta ocasionando así grandes daños en la economía de los agricultores.

Las principales plagas que afectan este tipo de cultivo son:

Nematodos: Los nemátodos producen modulaciones en la raíz que afecta al crecimiento de la planta.

Ácaros: Son pequeñas arañas que se puede ver en el fruto ya que secretan una sustancia que daña al mismo, tornándolo de un color marrón y provocando que estas se pudran.

Pulgón: Son pequeños insectos que se ubican en las hojas, succionan la sabia y provocan que las hojas se enrollen y se tornen de color amarillo y con el paso del tiempo color marrón.

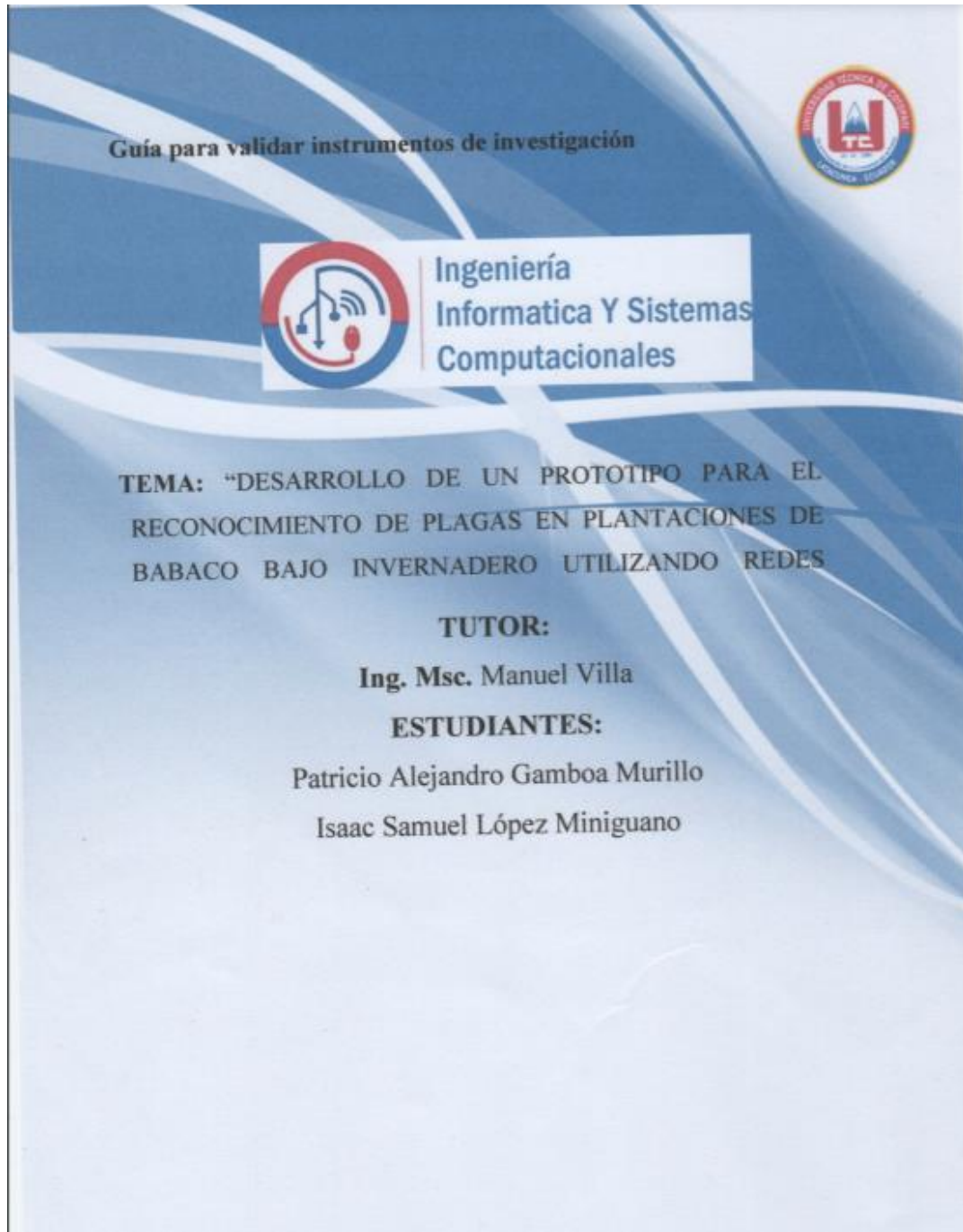
 180130862-6.

Sr. Héctor Mecías

8.3.1. Anexo: Cedula del agricultor Héctor Mecías



8.4. Anexo: Instrumento aplicado a expertos



1. Tema

Desarrollar un prototipo para el reconocimiento de plagas en plantaciones de babaco bajo invernadero mediante redes neuronales

2. Resumen

En este trabajo se analiza las plagas que atacan al babaco, por medio de un entrenamiento de redes neuronales con el procesamiento de imágenes capturadas por el grupo de investigación. Este proyecto vertical está diseñado en beneficio de los agricultores del cantón Patate Provincia de Tungurahua. El cual se utilizó Python para crear la arquitectura Keras mediante Google Colab el cual nos permite trabajar con el lenguaje de programación ya dicho a la vez utilizamos las librerías TensorFlow y Open CV para el entrenamiento y detección del nivel de afectación de las plagas (ácaros y pulgones).

3. Objetivos: General y específicos

Objetivo general:

Desarrollar un prototipo para el reconocimiento de plagas en plantaciones de babaco bajo invernadero, utilizando redes neuronales.

Objetivos específicos:

- Diseñar un esquema de una solución integral para la detección de plagas en plantaciones de babaco bajo invernadero, para la delimitación del prototipo vertical a ser desarrollado.
- Analizar los resultados de desempeño de modelos de redes neuronales empleados en proyectos de visión artificial, mediante procesos de revisión bibliográfica, para la selección de un modelo adecuado al contexto de estudio.
- Entrenar la red neuronal, utilizando imágenes de plagas en plantaciones de babaco bajo invernadero, para la realización de pruebas de clasificación.

- Evaluar el modelo diseñado, mediante la ejecución de pruebas, para la determinación de su nivel de efectividad en diferentes escalas de ataque de la plaga, y para la comparación con los resultados de otras investigaciones.

4. Hipótesis

La efectividad generada por el uso de redes neuronales en el reconocimiento de plagas en plantas de babaco bajo invernadero es aceptable respecto al proceso de reconocimiento manual. (T de Student)

Ho: No existe una diferencia significativa en la efectividad generada por el uso de redes neuronales en el reconocimiento de plagas en plantas de babaco bajo invernadero respecto al proceso de reconocimiento manual.

Ha: Existe una diferencia significativa en la efectividad generada por el uso de redes neuronales en el reconocimiento de plagas en plantas de babaco bajo invernadero respecto al proceso de reconocimiento manual.

Hipótesis	Variables	Indicadores	Técnicas
La efectividad generada por el uso de redes neuronales en el reconocimiento de plagas en plantas de babaco bajo invernadero es aceptable respecto al proceso de reconocimiento manual.	Variable Dependiente	• Redes neuronales	• Pruebas de prototipo
	Efectividad en el Reconocimiento	• Reconocimiento manual (experto)	• Observación
	Variable Independiente	• Tipo de plaga	• Observación
	Plagas en plantas de babaco bajo invernadero	• Escala de ataque	

Tabla 1. Hipótesis

5. Formulario

Indique el grado frente a las siguientes indagaciones como usted considere: 1 = muy en desacuerdo 2 = en desacuerdo 3 = en desacuerdo más que en acuerdo 4 = de acuerdo más que en desacuerdo 5 = de acuerdo, 6 = muy de acuerdo	Grado de acuerdo					
	1	2	3	4	5	6

Pregunta n. 1

Indique su grado de acuerdo frente a las siguientes afirmaciones: (1 = muy en desacuerdo; 2 = en desacuerdo; 3 = en desacuerdo más que en acuerdo; 4 = de acuerdo más que en desacuerdo; 5 = de acuerdo; 6 = muy de acuerdo)	Grado de acuerdo					
	1	2	3	4	5	6
¿Considera usted que el ataque de las plagas es un aspecto a tomar en cuenta durante la producción del babaco?						X
¿Por qué? <i>CAUSA AÑOS ECONÓMICOS, ADEMÁS ES MUY PRESUDICIAL PARA LA SALUD.</i>						

Pregunta n. 2

Indique su grado de acuerdo frente a las siguientes afirmaciones: (1 = muy en desacuerdo; 2 = en desacuerdo; 3 = en desacuerdo más que en acuerdo; 4 = de acuerdo más que en desacuerdo; 5 = de acuerdo; 6 = muy de acuerdo)	Grado de acuerdo					
	1	2	3	4	5	6
¿Considera usted que la red neuronal entrenada ayudó en la clasificación de las imágenes para la evaluación correspondiente?						X
¿Por qué? PORQUE ES UNA TÉCNICA EFICIENTE YA QUE BORRA DATOS MAS CORRECTOS Y RECIEN.						

Pregunta n. 3

Indique su grado de acuerdo frente a las siguientes afirmaciones: (1 = muy en desacuerdo; 2 = en desacuerdo; 3 = en desacuerdo más que en acuerdo; 4 = de acuerdo más que en desacuerdo; 5 = de acuerdo; 6 = muy de acuerdo)	Grado de acuerdo					
	1	2	3	4	5	6
¿La ejecución de pruebas del entrenamiento es capaz de detectar el nivel de ataque de las plagas?						X
¿Por qué? ACORDA A LOS RESULTADOS SE DIO O PROPITO EN UN BUEN PORCENTAJE EL NIVEL DE APORTACIÓN DE LAS ADOS Y FANTOS.						

Pregunta n. 4

Indique su grado de acuerdo frente a las siguientes afirmaciones: (1 = muy en desacuerdo; 2 = en desacuerdo; 3 = en desacuerdo más que en acuerdo; 4 = de acuerdo más que en desacuerdo; 5 = de acuerdo; 6 = muy de acuerdo)	Grado de acuerdo					
	1	2	3	4	5	6
¿El entrenamiento de la red neuronal es capaz de mostrar el tipo de plaga que es?						X
¿Por qué? SEGUN LOS RESULTADOS DEL ENTRENAMIENTO SI SE PUEDE REVELAR QUE TIPO ES.						


Pregunta n. 5

Indique su grado de acuerdo frente a las siguientes afirmaciones: (1 = muy en desacuerdo; 2 = en desacuerdo; 3 = en desacuerdo más que en acuerdo; 4 = de acuerdo más que en desacuerdo; 5 = de acuerdo; 6 = muy de acuerdo)	Grado de acuerdo					
	1	2	3	4	5	6
¿Considera usted que la investigación realizada ayudará en las labores de campo a los agricultores de babaco?						X
¿Por qué? LOS AYUDOS TECNOLÓGICOS SON DE GRAN APOYO, PARA LLEVAR UN CONTROL ADECUADO EN LA DETECCIÓN DE PLAGAS, PARA MEJORAR PROCESOS AGRÍCOLAS.						

		Evaluación general del cuestionario			
		Excelente	Buena	Regular	Deficiente
Validez de contenido del cuestionario					
Observaciones y recomendaciones en general del cuestionario.					
Motivos por los que se considera no adecuada	Recolección de fotografías.				
Motivos por los que se considera no pertinente					
Propuestas de mejora (modificación, sustitución o supresión)	POR PARTE DE OTRO GRUPO DE INVESTIGADORES IMPLEMENTAR UNA APLICACIÓN O INTERFAZ QUE MUESTRE RESULTADOS DE UNA MEJOR MONEDA.				

Identificación del experto

Nombres y apellidos	LUIS HECTOR MECIAS ZESANO
Filiación (ocupación, grado académico y lugar de trabajo):	AGRICULTOR.
e-mail	
Teléfono o celular	032 859 333
Fecha de la	

validación (día, mes y año):	27 / OCTUBRE / 2020
Firma	 180130962-6

Muchas gracias por su valiosa contribución a la validación de este cuestionario.

8.4.1. Anexo: Instrumento aplicado a experto en redes neuronales

Pregunta n. 1

Indique su grado de acuerdo frente a las siguientes afirmaciones: (1 = muy en desacuerdo; 2 = en desacuerdo; 3 = en desacuerdo más que en acuerdo; 4 = de acuerdo más que en desacuerdo; 5 = de acuerdo; 6 = muy de acuerdo)	Grado de acuerdo					
	1	2	3	4	5	6
¿Considera usted que el ataque de las plagas es un aspecto a tomar en cuenta durante la producción del babaco?						✓
¿Por qué? Ayuda al buen crecimiento de la producción del babaco.						

Pregunta n. 2

Indique su grado de acuerdo frente a las siguientes afirmaciones: (1 = muy en desacuerdo; 2 = en desacuerdo; 3 = en desacuerdo más que en acuerdo; 4 = de acuerdo más que en desacuerdo; 5 = de acuerdo; 6 = muy de acuerdo)	Grado de acuerdo					
	1	2	3	4	5	6
¿Considera usted que la red neuronal entrenada ayudó en la clasificación de las imágenes para la evaluación correspondiente?						✓
¿Por qué? Ayuda a la identificación temprana del tipo de plagas clasificándolos de muy buena manera las imágenes obtenidas.						

Pregunta n. 3

Indique su grado de acuerdo frente a las siguientes afirmaciones: (1 = muy en desacuerdo; 2 = en desacuerdo; 3 = en desacuerdo más	Grado de acuerdo

que en acuerdo; 4 = de acuerdo más que en desacuerdo; 5 = de acuerdo; 6 = muy de acuerdo)	1	2	3	4	5	6
¿La interfaz es capaz de detectar el tipo de plaga?						/
¿Por qué? La interfaz es capaz de elaborar una predicción importante en el momento de identificar el tipo de plaga.						

Pregunta n. 4

Indique su grado de acuerdo frente a las siguientes afirmaciones: (1 = muy en desacuerdo; 2 = en desacuerdo; 3 = en desacuerdo más que en acuerdo; 4 = de acuerdo más que en desacuerdo; 5 = de acuerdo; 6 = muy de acuerdo)	Grado de acuerdo					
	1	2	3	4	5	6
¿La interfaz es capaz de mostrar el tipo de plaga que es?						/
¿Por qué? Presenta el tipo de plaga que la imagen contiene.						

Pregunta n. 5


Indique su grado de acuerdo frente a las siguientes afirmaciones: (1 = muy en desacuerdo; 2 = en desacuerdo; 3 = en desacuerdo más que en acuerdo; 4 = de acuerdo más que en desacuerdo; 5 = de acuerdo; 6 = muy de	Grado de acuerdo					
	1	2	3	4	5	6

acuerdo)									
¿Considera usted que la investigación realizada ayuda a los agricultores de babaco a llevar un mejor control en la detección de plagas en los cultivos?									

¿Por qué? Tardaron menos perdidos al momento del cultivo, ya que identificación de una manera eficaz las plagas, el cual ayudaría a identificar un remedio eficaz a tiempo.

Evaluación general del cuestionario				
	Excelente	Buena	Regular	Deficiente
Validez de contenido del cuestionario				
Observaciones y recomendaciones en general del cuestionario.				
Motivos por los que se considera no adecuada				
Motivos por los que se considera no pertinente				
Propuestas de mejora (modificación, sustitución o supresión)	Desde menú puntos de vista deben implementar dentro de la interfaz móvil una herramienta el cual me ayude a capturar imágenes en tiempo real.			

Identificación del experto

Nombres y apellidos	Gonzalo Ramón Balvender Escobar.
Filiación (ocupación, grado académico y lugar de trabajo):	Especialista en redes neurales y en aplicaciones móviles y web. Ingeniería en informática.
e-mail	gonzalovalvender08@gmail.com
Teléfono o celular	097994676
Fecha de la validación (día, mes y año):	21 / 08 / 2021
Firma	 130398849

Muchas gracias por su valiosa contribución a la validación de este cuestionario.

8.5. Anexo: Gastos indirectos

GASTOS INDIRECTOS				
SERVICIOS BÁSICOS	\$25,00	\$25,00	\$25,00	\$25,00
PASAJES	\$8,00	\$8,00	\$8,00	\$8,00
TOTAL	\$132			

Tabla 9. Gastos indirectos

8.6. Anexo: Gastos directos

RESULTADOS/ACTIVIDADES	MESES				
	1er	2do	3ro	4to	Total
RECURSOS MATERIALES					
Materiales y suministros de oficina	\$5	-	\$50	-	\$55
Impresiones	\$5	-	\$25	-	\$30
Anillados	-	-	\$10,00	\$10,00	\$20
Encuadernación	-	-	-	\$80,00	\$80
RECURSOS TECNOLÓGICOS					
Internet	\$20,00	\$20,00	\$20,00	\$20,00	\$80
Datos móviles	\$5,00	\$5,00	\$5,00	-\$5,00	\$20,00
Total, gastos Directos					\$285,00

Tabla 10. Gastos directos

8.7. Anexo: Fotografías de toma de muestras por parte del grupo investigador.



Figura 26. Fotos del grupo investigador