



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN SOFTWARE**

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN SOFTWARE**

**TEMA: DESARROLLO DE UN SISTEMA SOFTWARE DE APRENDIZAJE
PROFUNDO QUE OPTIMICE EL DIAGNÓSTICO DE PLAGAS EN
HOJAS DE CULTIVOS DE PAPAS EN EL SECTOR LA LIBERTAD
PARROQUIA TOACASO PROVINCIA DE COTOPAXI**

**AUTORES: CHOLOQUINGA COCHA, LUIS GERARDO
REYES NEGRETE, ALEX OMAR**

DIRECTOR: ING. GARCÉS GUAYTA, LUCAS ROGERIO

LATACUNGA

2020



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN SOFTWARE

CERTIFICADO DEL DIRECTOR

Certifico que el trabajo de titulación, “**DESARROLLO DE UN SISTEMA SOFTWARE DE APRENDIZAJE PROFUNDO QUE OPTIMICE EL DIAGNÓSTICO DE PLAGAS EN HOJAS DE CULTIVOS DE PAPAS EN EL SECTOR LA LIBERTAD PARROQUIA TOACASO PROVINCIA DE COTOPAXI**” fue realizado por los señores: **Choloquina Cocha, Luis Gerardo y Reyes Negrete, Alex Omar**, el mismo que ha sido revisado en su totalidad, analizado por la herramienta de verificación de similitud de contenido; por lo tanto, cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Latacunga, 15 de enero del 2020



Ing. Lucas Rogério, Garcés Guayta
C.C. 1802665800



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN SOFTWARE

AUTORÍA DE RESPONSABILIDAD

Nosotros, **Choloquina Cocha, Luis Gerardo y Reyes Negrete, Alex Omar**, declaramos que el contenido, ideas y criterios del trabajo de titulación: **“DESARROLLO DE UN SISTEMA SOFTWARE DE APRENDIZAJE PROFUNDO QUE OPTIMICE EL DIAGNÓSTICO DE PLAGAS EN HOJAS DE CULTIVOS DE PAPAS EN EL SECTOR LA LIBERTAD PARROQUIA TOACASO PROVINCIA DE COTOPAXI”**, es de nuestra autoría y responsabilidad, cumpliendo con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Consecuentemente el contenido de la investigación mencionada es veraz.

Latacunga, 15 de enero del 2020


Choloquina Cocha, Luis Gerardo
C.C. 0504480617


Reyes Negrete, Alex Omar
C.C. 0503895237



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN SOFTWARE

AUTORIZACIÓN

Nosotros, **Choloquina Cocha, Luis Gerardo y Reyes Negrete, Alex Omar**, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **“DESARROLLO DE UN SISTEMA SOFTWARE DE APRENDIZAJE PROFUNDO QUE OPTIMICE EL DIAGNÓSTICO DE PLAGAS EN HOJAS DE CULTIVOS DE PAPAS EN EL SECTOR LA LIBERTAD PARROQUIA TOACASO PROVINCIA DE COTOPAXI”**, en el Repositorio Institucional, cuyo contenido, ideas y criterios son de nuestra responsabilidad.

Latacunga, 15 de enero del 2020

Choloquina Cocha, Luis Gerardo

C.C. 0504480617

Reyes Negrete, Alex Omar

C.C. 0503895237

DEDICATORIA

Una vez culminado este proyecto de investigación, quiero dedicar este trabajo a toda mi familia que me han brindado su apoyo incondicional y enseñado valores como humildad y respeto, para poder alcanzar mis metas, en especial a mis padres los cuales siempre me han brindado sus consejos, palabras de aliento para crecer día a día como persona, y siempre guiándome en todo el transcurso de mis estudios.

Sobre todo, se lo dedico a dios por su amor incondicional por acompañarme y guiarme siempre en mi camino, por la salud y vida que nos brinda a mí y a toda mi familia.

Alex Reyes

DEDICATORIA

Dedico este proyecto de titulación a mi familia, en especial a mis padres por el enorme esfuerzo realizado, al permitirme estudiar y culminar una carrera universitaria, gracias a su paciencia, cariño y apoyo condicional, que me han permitido al día de hoy cumplir una meta más en mi crecimiento personal.

A mis hermanos por su cariño y apoyo incondicional, por estar conmigo durante toda mi etapa de formación académica. A toda mi familia gracias a sus consejos y palabras de aliento, al estar conmigo en los buenos y malos momentos.

Luis Choloquina

AGRADECIMIENTO

En primer lugar, agradezco a Dios por darme salud y vida, por guiarme siempre por el camino del bien, darme fuerzas y sabiduría en momentos difíciles de mi vida a la vez permitirme cumplir una meta más trazada en mi vida.

A mis padres, a quienes les debo todo, gracias por todos los sacrificios que han realizado para poder lograr mis objetivos en la vida, por enseñarme grandes valores que me han ayudado a crecer como persona. Por estar apoyándome siempre en los buenos y malos momentos y enseñándome a superar las situaciones difíciles.

A mis amigos por ser grandes personas, con los cuales compartimos buenos momentos y experiencias únicas dentro y fuera de las aulas, por brindarme su apoyo en situaciones difíciles a lo largo de la carrera.

Finalmente, pero no menos importante agradezco a la Universidad de las Fuerzas Armadas ESPE extensión Latacunga por ofrecer los recursos necesarios durante todo el transcurso de la carrera, en especial agradezco a mi tutor de tesis Ingeniero Lucas Garcés por apoyarnos y brindarme sus conocimientos para culminar con éxito el desarrollo de este proyecto de investigación.

Alex Reyes

AGRADECIMIENTO

En primer lugar, agradezco a Dios, por guiarme por el camino del bien, brindarme salud, inteligencia y paciencia durante esta etapa de formación académica, permitiéndome cumplir con un objetivo más en mi crecimiento personal.

A mis padres, gracias por el enorme esfuerzo realizado para que yo pueda culminar un ciclo más en mi etapa de crecimiento personal, por estar conmigo en los buenos y malos momentos, gracias a ustedes soy lo que soy y hasta donde he llegado.

A mis hermanos, gracias por estar ahí cuando más los necesite con sus palabras de aliento, cariño y paciencia. A mi familia en general no me queda más que decir que muchísimas gracias por sus consejos y ayudarme a mantenerme de quien soy, de donde vengo y hacia dónde voy.

Finalmente, pero no menos importante a la Universidad de las Fuerzas Armadas ESPE extensión Latacunga, por permitirme estudiar en una de las mejores universidades del país y brindarme todas las herramientas necesarias para culminar mi carrera universitaria.

Luis Choloquina

ÍNDICE DE CONTENIDOS

CARÁTULA

| | |
|---------------------------------|-------|
| CERTIFICADO DEL DIRECTOR | i |
| AUTORÍA DE RESPONSABILIDAD..... | ii |
| AUTORIZACIÓN | iii |
| DEDICATORIA..... | iv |
| AGRADECIMIENTO | vi |
| ÍNDICE DE CONTENIDOS..... | viii |
| ÍNDICE DE FIGURAS | xiv |
| ÍNDICE DE TABLAS | xvi |
| RESUMEN | xviii |
| ABSTRACT..... | xix |

CAPÍTULO I

DEFINICIÓN DEL PROBLEMA

| | | |
|-------|----------------------------------|---|
| 1.1 | Planteamiento del Problema..... | 1 |
| 1.2 | Formulación del Problema | 3 |
| 1.3 | Antecedentes..... | 3 |
| 1.4 | Justificación e importancia..... | 6 |
| 1.5 | Objetivos..... | 7 |
| 1.5.1 | Objetivo General..... | 7 |
| 1.5.2 | Objetivos Específicos | 7 |
| 1.6 | Meta..... | 8 |

| | | |
|-------|-------------------------------------|---|
| 1.7 | Hipótesis | 8 |
| 1.8 | Variables de la investigación | 8 |
| 1.8.1 | Variable dependiente | 8 |
| 1.8.2 | Variable Independiente | 8 |
| 1.9 | Indicadores | 9 |

CAPÍTULO II

MARCO TEÓRICO

| | | |
|--------|---|----|
| 2.1. | Introducción | 10 |
| 2.2. | Antecedentes Históricos | 10 |
| 2.2.1. | Evolución de métodos y técnicas de diagnóstico de plagas en cultivos de papas..... | 10 |
| 2.3. | Antecedentes Conceptuales y Referenciales | 17 |
| 2.3.1. | Caracterización Gnoseológica del proceso de diagnóstico de plagas en cultivos de papas | 17 |
| a. | La papa..... | 17 |
| b. | Plagas..... | 17 |
| c. | Proceso de diagnóstico de plagas | 18 |
| d. | Grupos de plagas en Cultivos de Papas..... | 20 |
| e. | Principales plagas que Amenazan los Cultivos de Papas | 21 |
| f. | Instituciones orientadas a la investigación y regulación del sector agropecuario..... | 22 |
| 2.3.2. | Caracterización tecnológica de los métodos y técnicas de diagnóstico de plagas en cultivos de papas..... | 25 |

| | | |
|------|--|----|
| a. | Método de detección sintomatológica visual..... | 25 |
| b. | Tecnologías aplicadas a los métodos de diagnóstico de plagas o enfermedades..... | 26 |
| b.1. | Tecnologías de Inteligencia artificial aplicadas a la detección de plagas en cultivos..... | 29 |
| b.2. | Herramientas para el desarrollo de sistema software de aprendizaje profundo..... | 41 |
| c. | Metodología Mobile-D..... | 47 |
| c.1. | Introducción | 47 |
| c.2. | Principios | 48 |
| c.3. | Fases del Mobile-D..... | 48 |
| c.4. | Ventajas..... | 50 |
| c.5. | Desventajas | 50 |
| 2.4. | Antecedentes Contextuales | 50 |

CAPÍTULO III

DESARROLLO DEL SISTEMA SOFTWARE DE APRENDIZAJE PROFUNDO QUE OPTIMICE EL DIAGNÓSTICO DE PLAGAS EN HOJAS DE CULTIVOS DE PAPAS

| | | |
|------|----------------------------------|----|
| 3.1. | Introducción al capítulo | 54 |
| 3.2. | Modelos de redes neuronales..... | 55 |

| | | |
|--------|--|----|
| 3.2.1. | Análisis comparativo entre arquitecturas de redes neuronales..... convolucionales | 56 |
| 3.3. | Proceso de desarrollo del modelo de red neuronal convolucional para el..... diagnóstico de plagas en cultivos de papas..... | 58 |
| 3.4. | Desarrollo del modelo de red neuronal convolucional | 58 |
| 3.4.1. | Recolección de imágenes (dataset)..... | 58 |
| 3.4.2. | Preparación del entorno de trabajo..... | 61 |
| 3.4.3. | Configuración de la API de detección de objetos de Tensorflow | 61 |
| 3.4.4. | Etiquetado de imágenes (Dataset) | 62 |
| 3.4.5. | Preparación de los archivos de configuración del modelo de red neuronal.... | 67 |
| 3.4.6. | Entrenamiento y pruebas del modelo de red neuronal | 71 |
| 3.4.7. | Validación del modelo de red neuronal..... | 76 |
| 3.4.8. | Convertir el modelo de red neuronal a un formato compatible para | 79 |
| | dispositivos móviles | |
| 3.5. | Desarrollo de la aplicación móvil | 80 |
| 3.5.1. | Exploración | 80 |
| a. | Establecimiento de los Stakeholders | 81 |
| b. | Establecimiento del Proyecto..... | 81 |
| 3.5.2. | Inicialización | 82 |
| a. | Requisitos Iniciales | 82 |
| b. | Análisis de requerimientos iniciales | 82 |
| c. | Pila del Producto..... | 83 |
| d. | Módulos de la aplicación móvil | 84 |
| e. | Limitaciones..... | 84 |

| | | |
|--------|---|----|
| f. | Supuestos y Dependencias | 84 |
| g. | Planificación Inicial | 84 |
| h. | Análisis de requerimientos y pre-requisitos | 87 |
| i. | Planificación de fases | 88 |
| j. | Diseño de la aplicación | 89 |
| k. | Arquitectura de la aplicación | 89 |
| l. | Descripción de las Interfaces de Usuario | 90 |
| 3.5.3. | Producción, estabilización y pruebas..... | 97 |
| a. | Implementación de la aplicación | 97 |
| a.1. | Estándares de codificación | 97 |
| b. | Pruebas de la aplicación móvil | 97 |
| b.1. | Pruebas de interfaz..... | 98 |
| b.2. | Pruebas de aceptación | 99 |

CAPITULO IV

VALIDACIÓN Y ANÁLISIS DE LA APLICACIÓN

| | | |
|--------|--|-----|
| 4.1. | Introducción | 101 |
| 4.2. | Procesamiento de los resultados | 101 |
| 4.2.1. | Análisis y resultados de la aplicación del instrumento a los técnicos | |
| | agrónomos del Ministerio de Agricultura y Ganadería (MAG)..... | 103 |
| 4.2.2. | Análisis y resultados de la aplicación del instrumento a los expertos en | |
| | cultivos de papas. | 105 |
| 4.3. | Prueba de hipótesis mediante chi cuadrado | 106 |

| | | |
|--------|---|-----|
| 4.3.1. | Análisis de la hipótesis general..... | 106 |
| 4.3.2. | Cálculo de frecuencias esperadas, correspondientes a cada frecuencia observada. | 107 |
| a. | Frecuencia Observada | 107 |
| 4.3.3. | Cálculo del valor de chi cuadrado..... | 109 |
| 4.3.4. | Cálculo del valor crítico de chi cuadrado | 111 |
| 4.3.5. | Comparación entre el valor esperado y el valor crítico | 112 |
| 4.3.6. | Conclusiones de los resultados presentados..... | 113 |

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

| | | |
|---|-----------------------|------------|
| 5.1. | Introducción | 115 |
| 5.2. | Conclusiones | 115 |
| 5.3. | Recomendaciones | 117 |
| REFERENCIAS BIBLIOGRÁFICAS | | 119 |
| ANEXOS | | 132 |

ÍNDICE DE FIGURAS

| | |
|--|----|
| Figura 1. Proceso de Diagnóstico de plagas | 19 |
| Figura 2. Aplicación móvil para detectar enfermedades en la mandioca | 28 |
| Figura 3. Elementos de una red neuronal..... | 31 |
| Figura 4. Red Neuronal Convolutiva | 34 |
| Figura 5. Arquitectura de una red neuronal convolutiva..... | 40 |
| Figura 6. Desarrollo del modelo de red neuronal..... | 58 |
| Figura 7. Psílido de la punta morada | 59 |
| Figura 8. Bactericera cockerelli..... | 59 |
| Figura 9. Mosca minadora de la papa | 60 |
| Figura 10. Mosca minadora adulta | 60 |
| Figura 11. API de detección de objetos de TensorFlow | 62 |
| Figura 12. Etiquetado del insecto de la plaga punta morada..... | 63 |
| Figura 13. Etiquetado del insecto de la plaga punta morada..... | 63 |
| Figura 14. Etiquetado del insecto de la plaga mosca minadora..... | 64 |
| Figura 15. Información del resultado de etiquetado de imagen | 64 |
| Figura 16. Archivos XML del conjunto de datos..... | 65 |
| Figura 17. Archivos CSV del conjunto de datos..... | 66 |
| Figura 18. Archivos TFRecord | 67 |
| Figura 19. Modelo pre-entrenado con arquitectura MobileNet..... | 67 |
| Figura 20. Archivos TFRecord de entrenamiento y pruebas para la red neuronal..... | 68 |
| Figura 21. Archivo de configuración del modelo | 69 |

| | |
|--|-----|
| Figura 22. Etiquetas del modelo de red neuronal | 70 |
| Figura 23. Código para vincular el archivo Colaboratory a Google Drive | 71 |
| Figura 24. Código para ejecutar el entrenamiento del modelo | 72 |
| Figura 25. Inicio de entrenamiento del modelo | 73 |
| Figura 26. Disminución de la función de pérdida | 73 |
| Figura 27. Proceso de entrenamiento..... | 74 |
| Figura 28. Función de pérdida del modelo | 75 |
| Figura 29. Comando para generar el archivo congelado..... | 76 |
| Figura 30. Archivo congelado frozen_inference_graph | 77 |
| Figura 31. Archivo detect. tflite | 80 |
| Figura 32. Diseño de la aplicación..... | 89 |
| Figura 33. Arquitectura de la aplicación..... | 90 |
| Figura 34. Pantalla de prueba de interfaz de la aplicación | 98 |
| Figura 35. Criterio de aceptación del sistema software (aplicación móvil) por parte | |
| de los técnicos agrónomos del ministerio de agricultura y ganadería | |
| (MAG)..... | 104 |
| Figura 36. Criterio de aceptación del sistema software (aplicación móvil) por parte | |
| de los expertos en cultivos de papas..... | 105 |
| Figura 37. Prueba de Chi Cuadrado Dependiente..... | 113 |

ÍNDICE DE TABLAS

| | |
|--|----|
| Tabla 1 Modelos pre-entrenados para aprendizaje profundo de TensorFlow | 36 |
| Tabla 2 Criterios para seleccionar el modelo de red neuronal adecuado para el uso..... en aplicaciones móviles. “Cuadro comparativo de redes neuronales” | 57 |
| Tabla 3 Nombres de etiquetas para el entrenamiento del modelo | 70 |
| Tabla 4 Función de pérdida por cada 10000 iteraciones | 74 |
| Tabla 5 Pruebas realizadas al modelo de red neuronal | 78 |
| Tabla 6 Requisitos Iniciales..... | 83 |
| Tabla 7 Pila del Producto | 83 |
| Tabla 8 Historia de Usuario: Detectar posibles amenazas de plagas en hojas de..... cultivos de papas..... | 85 |
| Tabla 9 Historia de Usuario: Información de las plagas existentes en cultivos de papas..... | 85 |
| Tabla 10 Historia de Usuario: Información de insecticidas existentes para el..... control de plagas. | 86 |
| Tabla 11 Historia de Usuario: Información sobre el proceso de Diagnóstico de plagas en cultivos de papas..... | 86 |
| Tabla 12 Planificación de fases..... | 88 |
| Tabla 13 Icono de la aplicación..... | 90 |
| Tabla 14 Pantallas de bienvenida | 91 |
| Tabla 15 Menú Principal | 93 |
| Tabla 16 Pantalla de Inicio..... | 94 |

| | |
|--|-----|
| Tabla 17 <i>Pantalla Diagnóstico en Tiempo Real</i> | 94 |
| Tabla 18 <i>Pantalla Control de Plagas</i> | 95 |
| Tabla 19 <i>Pantalla Información de Plagas</i> | 95 |
| Tabla 20 <i>Pantalla Proceso Diagnóstico Plagas</i> | 96 |
| Tabla 21 <i>Prueba de aceptación Detectar posibles amenazas de plagas en hojas de cultivos de papas</i> | 99 |
| Tabla 22 <i>Prueba de aceptación Información de las plagas existentes en cultivos de..... papas</i> | 99 |
| Tabla 23 <i>Prueba de aceptación Información de insecticidas existentes para el control de plagas</i> | 100 |
| Tabla 24 <i>Prueba de aceptación Información sobre el proceso de diagnóstico de plagas en cultivos de papas</i> | 100 |
| Tabla 25 <i>Resultados de aceptación de los técnicos agrónomos del ministerio de..... agricultura y ganadería (MAG)</i> | 104 |
| Tabla 26 <i>Resultados de aceptación de los expertos en cultivos de papas</i> | 105 |
| Tabla 27 <i>Variable Sistema Software de aprendizaje profundo (aplicación móvil)</i> | 107 |
| Tabla 28 <i>Variable Optimiza el diagnóstico de plagas en hojas de cultivos de papas..... en el sector la Libertada parroquia Toacaso provincia de Cotopaxi</i> | 107 |
| Tabla 29 <i>Frecuencia Esperada para las Variables Independiente y Dependiente</i> | 108 |
| Tabla 30 <i>Frecuencia esperada de las dos variables</i> | 109 |
| Tabla 31 <i>Valor de Chi Cuadrado</i> | 110 |
| Tabla 32 <i>Distribución Chi Cuadrado Crítico</i> | 112 |

RESUMEN

El presente proyecto está orientado al desarrollo de un sistema software (aplicación móvil) que optimice el diagnóstico de plagas en hojas de cultivos de papas, está enfocado en funcionar como una herramienta de ayuda en el diagnóstico de plagas para técnicos agrónomos especializados, como para pequeños y medianos agricultores que se dedican al cultivo de papas. La aplicación móvil presenta cuatro funcionalidades: La primera funcionalidad permite el diagnóstico de plagas en tiempo real a través de la cámara del dispositivo móvil, este diagnóstico está basado en una de las técnicas del campo del aprendizaje profundo denominado redes neuronales convolucionales con arquitectura MobileNet, esta red neuronal está entrenada y validada con 20000 imágenes, con la finalidad de detectar dos tipos de plagas: psílido de la punta morada y mosca minadora, esto hace posible dar un diagnóstico inmediato de las posibles amenazas de plagas que están afectando al cultivo. La segunda funcionalidad trata sobre el control de plagas, permite visualizar información sobre los productos que puede aplicar de acuerdo al ciclo de vida del cultivo en la que se encuentre. La tercera funcionalidad contiene información más relevante sobre las principales plagas que actualmente afectan los cultivos de papas. La última funcionalidad de la aplicación móvil trata sobre el proceso diagnóstico plagas que permite ver los pasos detallados sobre cómo hacer un diagnóstico adecuado en sus cultivos. El desarrollo de la aplicación móvil está basado en los procesos de ingeniería en software y mediante el uso de la metodología denominado Mobile-D.

PALABRAS CLAVES

- **SOFTWARE MÓVIL**
- **REDES NEURONALES**
- **MOBILENET**
- **REDES CONVOLUCIONALES**

ABSTRACT

The present project is oriented to the development of a software system (mobile application) that optimizes the diagnosis of pests in leaves of potato crops. It is focused on functioning as an aid tool in the diagnosis of pests for specialized agronomists, as well as for small and medium farmers who are dedicated to the cultivation of potatoes. The mobile application has four functionalities: The first functionality allows the diagnosis of pests in real time through the camera of the mobile device, this diagnosis is based on one of the techniques of the deep learning field called convolutional neural networks with MobileNet architecture, this neural network is trained and validated with 20000 images, with the purpose of detecting two types of pests: purple tip psyllid and leafminer fly, this makes possible to give an immediate diagnosis of the possible threats of pests that are affecting the crop. The second functionality deals with pest control, allowing you to visualize information about the products you can apply according to the life cycle of the crop you are in. The third functionality contains more relevant information about the main pests that currently affect potato crops. The last functionality of the mobile application deals with the pest diagnostic process that allows you to see the detailed steps on how to make a proper diagnosis on your crops. The development of the mobile application is based on software engineering processes and using the methodology called Mobile-D.

KEYWORDS

- **MOBILE SOFTWARE**
- **NEURAL NETWORKS**
- **MOBILENET**
- **CONVOLUTION NETWORKS**

CAPÍTULO I

DEFINICIÓN DEL PROBLEMA

1.1 Planteamiento del Problema

La papa por años ha sido un cultivo de gran prioridad en el Ecuador. Hoy en día, los agricultores del país siembran anualmente cerca de 29.532 hectáreas de este cultivo, distribuido a nivel nacional de la siguiente manera: Carchi 25.82%, Pichincha 27%, Tungurahua 13.87%, Chimborazo 12.54%, Cotopaxi 10.41% (Instituto Nacional de Estadística y Censos, 2017).

En la actualidad varias condiciones han favorecido a que el cultivo enfrente varios problemas que ponen en riesgo el bienestar económico de los agricultores y en general la seguridad alimentaria del país. Debido en parte al intenso uso de pesticidas, han surgido plagas, constituyéndose en problemas y amenazas graves (Instituto Nacional de Investigaciones Agropecuarias, 2002).

Según los Ing. Agro. Segundo Cando e Ing. Agro. Paul Chacón técnicos especializados en cultivos de papas del Ministerio de Agricultura, Ganadería, Acuacultura y Pesca (MAGAP), en la provincia de Cotopaxi los cantones con más siembras de papas son: Latacunga, Salcedo y Saquisilí. Las principales parroquias de cultivo de papas en el cantón Latacunga son: Tanicuchi, Toacaso, Mulalo, Ignacio Flores, Aláquez y Juan Montalvo.

En la parroquia Toacaso se siembran anualmente alrededor de 4500 a 5000 hectáreas de papas, las cuales están divididas en diferentes sectores, los de mayor producción de papas son La Libertad y San Francisco.

Las principales plagas que amenazan los cultivos en el sector la Libertad son: la mosca minadora de la papa y psilido de la punta morada, causando pérdidas económicas a los pequeños y medianos productores de la zona.

Así también otro de los problemas a considerar son la presencia de enfermedades causadas por virus y bacterias que si no se detecta a tiempo esto puede causar grandes pérdidas de cultivos.

Por lo tanto, el problema principal está en el “diagnóstico de plagas en las hojas de plantas de cultivos de papas, debido a que los agrónomos de la zona realizan el diagnóstico mediante detección sintomatológica visual ¹ lo que implica un alto grado de complejidad, debido a que las plagas presentan formas similares, incluso los agrónomos y fitopatólogos ² experimentados a menudo no logran diagnosticar con éxito las plagas específicas y, por lo tanto, llegan a conclusiones y tratamientos erróneos” (Konstantinos, 2018).

Debido a la afectación de estas plagas en los cultivos de la zona la Libertad se tiene una pérdida de producción alrededor de un 30% en los pequeños y medianos productores de papas.

¹ Identificación visual a través del ojo humano

² Personas que se dedican al estudio de enfermedades causadas por virus bacterias y hongos

El trabajo de investigación presentado en este proyecto tiene como finalidad el desarrollo de un sistema software de aprendizaje profundo que optimice el diagnóstico de plagas en hojas de cultivos de papas. La aplicación móvil consiste en dar un diagnóstico de las plagas que pueden estar amenazando a los cultivos de papas en tiempo real, mediante el diagnóstico los agrónomos podrán emplear un plan preventivo para el manejo de control de plagas en el sector la Libertad parroquia Toacaso provincia de Cotopaxi.

1.2 Formulación del Problema

Con toda la información detallada en el anterior punto, se formula la siguiente interrogante: ¿Cómo optimizar el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertad parroquia Toacaso provincia de Cotopaxi?

1.3 Antecedentes

Las aplicaciones de Inteligencia Artificial relacionadas con el Aprendizaje Automático han logrado un crecimiento exponencial, lo que ha llevado al desarrollo de nuevas metodologías y modelos. Toman forma una nueva categoría, la de aprendizaje profundo (Lecun, Bengio, & Hinton, 2015).

En los últimos años, “el aprendizaje profundo, como los algoritmos de las redes neuronales convolucionales (CNN), las redes neuronales recurrentes (RNN) y las redes adversas generativas (GAN), se han estudiado y aplicado ampliamente en diversos campos, incluida la agricultura. Los investigadores en los campos de la agricultura a menudo usan marcos de software sin examinar suficientemente las ideas y mecanismos de una técnica” (Zhu, et al., 2018).

En la actualidad el aprendizaje profundo constituye una técnica reciente y moderna, para el procesamiento de imágenes y el análisis de datos, con prometedores resultados y un gran potencial. A medida que el aprendizaje profundo se ha aplicado con éxito en varios campos, uno de ellos es la agricultura aplicadas a diversos desafíos de la producción agrícola y alimentaria (Kamilaris, 2018).

La agricultura es una “industria donde los científicos e investigadores del aprendizaje profundo están trabajando, con agricultores para ayudarlos con sus productos. Los recursos para la agricultura como el agua, los fertilizantes, las tierras agrícolas, se están volviendo escasos cada día. El crecimiento de la urbanización ha causado una disminución en el área de tierras cultivables en los países en desarrollo desde mediados de los años ochenta. A medida que los recursos se vuelven más escasos, los científicos están utilizando el aprendizaje profundo para ayudar a los agricultores a hacer un uso mejor y más eficiente de los recursos restantes” (Hopkins, 2017).

Según Samiran (2018) expresa que “la agricultura es un área de aplicación particularmente compleja, cada área tiene sus propias características y clima, incluso eso cambia con el tiempo. Entonces, los agricultores buscan características de importancia como un mejor uso del agua y los nutrientes. La adopción del cambio en el clima y la resistencia a las enfermedades y plagas en un cultivo en crecimiento, requiere una gran cantidad de datos para analizar, teniendo en cuenta los cambios en tiempo real; Esto hace que el aprendizaje profundo sea un avance revolucionario que la agricultura

necesita, y con la ayuda de estos algoritmos, el fitomejoramiento ³ es cada vez más preciso y eficiente” (Samiran, 2018).

De igual manera Gallardo y Mantilla manifiestan que “la papa (*Solanum tuberosum*), desde varios años atrás es uno de los cultivos más importantes a nivel mundial. Como alimento para el hombre, tanto en los países en vías de desarrollo como en los desarrollados, el daño causado por las plagas a la papa en el campo es un problema a considerar para su producción” (Gallardo & Mantilla, 2008).

Con una superficie de 29.532 hectáreas cosechadas, una producción de 377.243 toneladas y un rendimiento de 12.77 toneladas/hectáreas, la papa es de primera importancia como fuente de alimento e ingresos entre los cultivos básicos de la provincia de Cotopaxi (Instituto Nacional de Estadística y Censos, 2017).

Según los Ing. Agro. Segundo Cando e Ing. Agro. Paul Chacón técnicos especializados en cultivos de papas del Ministerio de Agricultura, Ganadería, Acuacultura y Pesca (MAGAP), encargados de la parroquia Toacaso y Tanicuchí de la provincia de Cotopaxi respectivamente, se siembran anualmente alrededor de 4.500 a 5000 hectáreas de papas. En la parroquia de Toacaso el sector con más cultivos de papas es la Libertad en donde se ha determinado la presencia de plagas como: mosca minadora de la papa y psilido de la punta morada, siendo las principales amenazas en los cultivos de papas.

³ Es una técnica donde se cruzan distintas variedades de una misma especie como una solución para mejorar la producción

1.4 Justificación e importancia

Uno de los subcampos de la inteligencia artificial denominado aprendizaje profundo ya está siendo aplicado a la agricultura. Los países y las ciudades deberán preparar el camino, para que los ciudadanos y las organizaciones puedan aplicar las innovaciones tecnológicas para beneficio y el desarrollo de la productividad de la agricultura (Torres G. , 2018).

Los insecticidas mal aplicados durante el proceso de crecimiento de la papa han traído como consecuencia que este cultivo se vea afectado tanto en su rendimiento como en su calidad, poniendo en peligro el bienestar económico de los agricultores y la seguridad alimentaria del país, ya que este producto figura dentro de la canasta básica de los ecuatorianos (Cevallos & Mantilla, 2008).

El presente proyecto de investigación contribuirá con la optimización del diagnóstico de las plagas en hojas de cultivos de papas que actualmente son una de las causas de las mermas de la producción, por lo tanto, permitirá un manejo efectivo de las plagas, ya que con su correcto diagnóstico se podrán emplear los insecticidas adecuados para el control de las plagas, de este modo se apoyará al desarrollo técnico del cultivo de papa en el país.

Los beneficiarios del proyecto serán los pequeños y medianos productores de papas, así mismo los técnicos agrónomos especializados en cultivo de papas de la parroquia Toacaso.

A través del sistema software móvil desarrollado, los agricultores podrán detectar una posible plaga que pueda estar amenazando el cultivo en tiempo real y a través de eso emplear un plan preventivo para el control de sus cultivos incrementando su producción y disminuyendo las pérdidas causadas por las plagas. La toma de datos y pruebas se realizará en cultivos de papas del sector la Libertad de la parroquia Toacaso del cantón Latacunga provincia de Cotopaxi.

1.5 Objetivos

1.5.1 Objetivo General

Desarrollar un sistema software de aprendizaje profundo que optimice el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertad parroquia Toacaso de la Provincia de Cotopaxi.

1.5.2 Objetivos Específicos

- Elaborar el marco teórico sobre los diferentes algoritmos que existen en el campo del aprendizaje profundo, y sobre las plagas que amenazan los cultivos de papas a nivel de las hojas.
- Desarrollar el modelo de red neuronal mediante uno de los modelos de red neuronal pre-entrenados que existen en el campo del aprendizaje profundo.
- Desarrollar el sistema software móvil aplicando la metodología Mobile-D.
- Implementar el sistema software en el sector la Libertad parroquia Toacaso Provincia de Cotopaxi.
- Validar el sistema software mediante pruebas de funcionalidad y la valoración de un ingeniero agrónomo.

1.6 Meta

Desarrollo de un sistema software de aprendizaje profundo que optimice el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertad parroquia Toacaso Provincia de Cotopaxi.

1.7 Hipótesis

¿Si se desarrolla un sistema software de aprendizaje profundo entonces se optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertad parroquia Toacaso Provincia de Cotopaxi?

1.8 Variables de la investigación

1.8.1 Variable dependiente

Se optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertad parroquia Toacaso Provincia de Cotopaxi.

1.8.2 Variable Independiente

Se desarrolla un sistema software basado en aprendizaje profundo.

Conceptualización de la variable independiente

Un sistema software basado en aprendizaje profundo es un sistema informático dedicado al análisis de imágenes, aprende con los ejemplos, como lo harían los humanos, a partir de un conjunto de imágenes y datos registrados que representan características específicas de algún objeto. Durante el registro de la información sobre el objeto, el software desarrolla redes neuronales que pueden modelar su apariencia normal y sus defectos. Durante el tiempo de ejecución, el software localiza el objeto, extrae las

anomalías, hace una clasificación e incluso descifra caracteres difíciles de leer con rapidez, solidez y precisión.

1.9 Indicadores

- Número de plagas detectadas en tiempo real.
- Tiempo de análisis para el diagnóstico de plagas.
- Porcentaje de probabilidad de que sea la plaga correcta.
- Aplicar un plan un preventivo de plagas.
- Disminución de pérdida de producción de papas.

CAPÍTULO II

MARCO TEÓRICO

2.1. Introducción

En el presente capítulo se muestra información recopilada de los antecedentes históricos, antecedentes conceptuales, referenciales y los antecedentes contextuales que hacen referencia al proceso de diagnóstico de plagas en cultivos de papas y sobre los métodos y técnicas de detección que se aplican a este proceso.

De igual manera, se aborda las tecnologías y herramientas para el desarrollo del sistema software de aprendizaje profundo, así también se muestra la conceptualización de la metodología Mobile-D que será la guía para el desarrollo de la aplicación móvil de nuestro proyecto de investigación.

Por ello el marco teórico es de vital importancia debido a que aporta con el marco referencial necesario para delimitar el problema, plantear definiciones, fundamentar las hipótesis o afirmaciones que posteriormente tendrán que verificarse, e interpretar los resultados de estudio.

2.2. Antecedentes Históricos

2.2.1. Evolución de métodos y técnicas de diagnóstico de plagas en cultivos de papas.

Según el Instituto Nacional Autónomo de Investigaciones Agropecuarias (INIAP), en el documento “El cultivo de papa” manifiesta “La papa es uno de los principales cultivos a nivel mundial, produce más materia seca y proteína por hectáreas que los cereales, en

el caso de Ecuador representa la base de alimentación de gran parte de la población. El cultivo es fuertemente afectado por factores bióticos⁴ y abióticos⁵ que pueden ocasionar pérdidas significativas en la producción del cultivo” (Instituto Nacional de Investigaciones Agropecuarias, 2013).

Ivette Acuña, Rodrigo Bravo, María Paz Castro (2015), expresan que “En factores bióticos el cultivo de papas es fuertemente afectado por diversas plagas insectiles y enfermedades que pueden atacarla en distintas etapas fenológicas como productivas, y en diferentes órganos de la misma. A las problemáticas previas se suma la presencia de numerosas malezas asociadas, las cuales compiten por agua, luz, nutrientes y espacio con el cultivo de interés” (Ivette, et al., 2015).

Los técnicos y agricultores que manejan cultivos de papas están expuestos a la acción nociva de plagas y enfermedades, que si no se da un diagnóstico temprano puede llegar a ocasionar daños económicos al cultivo. Por ello recurren de forma sistemática⁶ a los tratamientos “por si acaso” o “bajo calendario”.

El proceso de diagnóstico de plagas en hojas de cultivos de papas se realiza mediante la técnica de detección sintomatológica visual, esto “implica un grado alto de complejidad, debido a que las plagas presentan formas similares, incluso los agrónomos y fitopatólogos⁷ experimentados a menudo no logran diagnosticar con éxito las plagas

⁴ Plagas y enfermedades que pueden amenazar a los cultivos de papas en diferentes etapas de crecimiento

⁵ Surgen a través de los problemas de humedad y fertilizantes químicos y físicos del entorno

⁶ Sistema o método con que se clasifica algo

⁷ Persona encargada del diagnóstico de plagas y enfermedades de un cultivo

específicas y, por lo tanto, llegan a conclusiones y tratamientos erróneos” (Ferentinos, 2018).

A continuación, se muestra la evolución que ha tenido las técnicas de diagnóstico de plagas en cultivos.

a. Etapa 1 (1976 – 2017): Detección sintomatológica visual

En el año de 1976 se aplicó la técnica de detección sintomatológica visual debido al problema principal de la papa que fue y sigue siendo la presencia de plagas y enfermedades que afectan los cultivos en diferentes zonas del país (Instituto Nacional de Investigaciones Agropecuarias, 1976).

La técnica mencionada consiste en visitar periódicamente un cultivo para medir la densidad y estimar la distribución de plagas o enfermedades, esto permite al productor observar su evolución y así mismo dar el seguimiento oportuno para evitar repercusiones en la producción del cultivo, para ello el técnico o el agricultor debe conocer todo con respecto a los problemas fitopatológicos⁸, en saber los problemas que afectan o le pueden afectar de manera inmediata y probable o llevar consigo un material de apoyo o alguna guía.

Según (Rodríguez, Vásquez, & Araya, 1995) en su guía de identificación y manejo integrado de plagas y enfermedades define que la detección sintomatológica visual “se realiza por medio de muestreos los cuales son una herramienta para determinar

⁸ Ciencia que estudia las causas y el desarrollo de las enfermedades y plagas en las plantas

características cuantificables de la población de una plaga o enfermedad dentro del cultivo, con la finalidad de definir la necesidad de alguna medida de control”.

En Ecuador el Instituto Nacional de Investigadores Agropecuarias (INIAP) es la organización encargada de investigar sobre las diferentes plagas y enfermedades que se pueden dar en los cultivos de papas. Los resultados obtenidos en la investigación son publicados como guías de apoyo para los técnicos y agricultores de cultivos de papas (Instituto Nacional de Investigaciones Agropecuarias, 1976).

La guía publicada en el año de 1976, contiene texto descriptivo de las diferentes plagas que amenazan los cultivos de papas, información detallada de las características, formas que presenta cada uno de estos insectos y en qué etapa del ciclo de vida del cultivo suele presentarse, mediante esta información se puede emplear un plan para erradicar la plaga y de esa manera evitar pérdidas económicas en los cultivos.

Con la guía se le hace más fácil dar un diagnóstico de una posible amenaza de plaga, ya que el técnico o el agricultor de acuerdo al material de apoyo debe buscar las características de las plagas en los cultivos.

En el año 2013 se publicó una guía que ya no contienen solo texto descriptivo, también está incluido imágenes fotográficas donde se puede visualizar las características y formas de las plagas. De esta manera el técnico o el agricultor en las visitas que realiza a los cultivos realizar una comparación entre la imagen de la guía y lo que el observa en ese instante y de esa manera se le hace más fácil determinar si existe presencia de plagas en sus cultivos.

El Instituto Nacional Autónomo de Investigaciones Agropecuarias, en el documento “Guía Fotográfica de las principales plagas del cultivo de papa en el Ecuador” informa de “una guía que contiene fotos y textos cortos, de las principales plagas que afectan el cultivo de papa en el Ecuador. Está diseñada para ser usada por facilitadores y técnicos de campo como material de apoyo en eventos de capacitación para agricultores, estudiantes, etc.” (Instituto Nacional de Investigaciones Agropecuarias, 2013).

Esta guía permite al agricultor tener conocimiento de los síntomas que presentan los cultivos, las formas de contagio, condiciones favorables para la plaga, época en la que aparece la plaga, formas de contagio y un manejo integrado para evitar la presencia de la plaga y esa manera la guía contribuye un aporte fundamental para los agrónomos y agricultores de diferentes zonas de cultivos de papas en el Ecuador.

b. Etapa 2 (2017- Actualidad): Inteligencia Artificial

En los últimos años el uso de técnicas de inteligencia artificial, han logrado un crecimiento exponencial aplicado ampliamente en diversos campos, incluida la agricultura con el fin de facilitar a los agricultores información confiable para sembrar, regar, fertilizar, diagnosticar plagas y enfermedades o cosechar cultivos de manera rápida y eficiente.

- **Aprendizaje Automático**

Es una técnica de inteligencia artificial que estudia como dotar a las máquinas de capacidad de aprendizaje, entendido como la generalización del conocimiento a partir de un conjunto de experiencias o datos, los cuales permiten que el programa aprenda y

pueda realizar tareas específicas como clasificar grandes cantidades de datos y reconocer patrones complejos, entre otras (Dot CSV, 2019).

En el año 2017 apareció Nuru⁹, una aplicación móvil de PlantVillage para detectar enfermedades de los cultivos en África, la Dra. Amanda Ramcharan empezó en el año 2017 a trabajar en el grupo de investigación de PlantVillage, de la universidad de Pensilvania, donde comenzó recolectando imágenes de enfermedades de las plantas para entrenar modelos de inteligencia artificial, con el objetivo de construir un modelo que pudiera implementarse en un teléfono inteligente, 6 meses después apareció Nuru una aplicación que hace uso de la técnica de aprendizaje automático para clasificar las enfermedades en las hojas de la mandioca, que es el sustento de alimentación a millones de personas en África occidental (Hsiao, 2018)

El proceso de diagnóstico consiste en acercar tu teléfono a una hoja específica, la examinas a través de la cámara del dispositivo móvil y si existe algún síntoma aparece un recuadro mencionando el nombre de la enfermedad que está afectando al cultivo, una vez realizado el diagnóstico, la aplicación también cuenta con una opción para recibir consejos acerca de las mejores soluciones, la aplicación no necesita de conexión inalámbrica o potencia de procesamiento remota; es decir funciona en poblados rurales (Hsiao, 2018).

- **Aprendizaje Profundo**

⁹ Es el nombre de la aplicación desarrollada, que significa luz en lengua swahili

El Deep Learning es una técnica que busca la resolución de problemas que plantea el mundo real. Un enfoque práctico que se lleva a cabo a partir de redes neuronales profundas que imitan el modo de funcionar de nuestro cerebro. Es un conjunto de algoritmos no lineales que pueden aplicarse para el tratamiento de datos y el reconocimiento de patrones. Cuando nos referimos a una forma no lineal estamos hablando de que las capas de las redes neuronales artificiales se apilan en una jerarquía que va desde las características de nivel de abstracción bajo hasta las de nivel de abstracción más compleja, es decir que nos permite entrenar una inteligencia artificial para predecir las salidas, teniendo en cuenta un conjunto de entradas, donde a través de las capas se van a ir reconociendo patrones desde líneas básicas hasta formas más complejas y colores (Platzi, 2017).

En el año 2019 se desarrolló un sistema para el diagnóstico de afecciones en plantas de cítricos, utilizando las técnicas de aprendizaje profundo y procesamiento de imágenes, el sistema se basa en un modelo de red neuronal convolucional mediante la arquitectura de clasificación de imágenes MobileNet, diseñada para dispositivos móviles (VELASCO ZAPATA, 2019).

Para obtener un diagnóstico de las posibles afecciones en los cítricos, se debe introducir una fotografía de la hoja afectada, para ello existen 3 posibilidades: capturar la fotografía con la cámara del dispositivo, lo más cerca posible enmarcando la totalidad del área afectada, cargar desde la galería de imágenes o indicar la dirección de Localizador Uniforme de Recursos (URL) de una imagen en internet. Los resultados o diagnóstico final aparecen luego de introducir la imagen, indicando el nombre de la afectación del

cítrico y el porcentaje de acierto con el que fue detectado, es posible que el diagnóstico no presente afecciones o la imagen no sea reconocida (VELASCO ZAPATA, 2019).

2.3. Antecedentes Conceptuales y Referenciales

2.3.1. Caracterización Gnoseológica del proceso de diagnóstico de plagas en cultivos de papas

a. La papa

La papa es uno de los productos más importantes dentro de la producción de la sierra ecuatoriana, el producto constituye una fuente de alimentación para miles de familias ecuatorianas. La papa se cultiva en alturas comprendidas entre los 2700 a 4300 msnm¹⁰ a lo largo de todo el callejón interandino, no obstante, las mayores producciones de papas se han reportado en climas fríos con temperaturas comprendidas entre 9 y 11 °C y en zonas con alturas alrededor de 3000 msnm (Yupangui, 2016).

b. Plagas

Las plagas y enfermedades son unos de los factores que más negativamente inciden en la producción agropecuaria y siendo el sector agropecuario de gran importancia (Sánchez, 2016).

En la práctica agrícola, las plagas se usan para calificar cualquier nivel o grado de un insecto nocivo, aun cuando este se encuentre en un nivel de población. Este término se usa de manera regular y generalizada para aplicar drásticas medidas que son desfavorables para el deterioro del agroecosistema (Sánchez, 2016).

¹⁰ Metros sobre el nivel del mar

Las plagas y otros insectos causan daño en los cultivos de papas al alimentarse de la planta, existen varios tipos de plagas que absorben los alimentos de la planta de diferentes formas, puede ser picando las hojas, chupando y masticando lo que impide el normal desarrollo de la planta, así como también existen otras insectos plagas que transportan el virus de diferentes plagas hacia otros cultivos, causando el contagio de varias cosechas que al final producen grandes pérdidas en los agricultores (Méndez & Inostroza, 2009).

c. Proceso de diagnóstico de plagas

Las plagas en los cultivos causan considerables pérdidas tanto económicas como en la calidad de la papa, es necesario aplicar un manejo efectivo de plagas para poder identificarlas y conocer las alternativas de manejo integrado, es necesario realizar investigaciones y muestreos en diferentes zonas productoras del país para tomarlas como recomendaciones futuras para aplicar planes preventivos de manejo de plagas (Instituto Nacional de Investigaciones Agropecuarias, 2002).

Las herramientas de control como los pesticidas son efectivas con algunos tipos de plagas, es necesario saber con exactitud cuáles son las plagas que están afectando al cultivo y cuáles tienen la posibilidad de aparecer. Una identificación visual precisa permitirá dar un seguimiento a las plagas para poder aplicar medidas de control efectivos (Méndez & Inostroza, 2009).

Una de las mejores formas de obtener información necesaria para tomar decisiones de control de plagas es un monitoreo del cultivo. Esto significa evaluar el estado de la planta, mediante muestreos y niveles de infestación de las plagas en los cultivos. La

inspección periódica y sistemática del cultivo es muy utilizado para conocer el número de individuos de la plaga y evaluar el daño producido a la planta, el registro de los muestreos obtenidos es de gran importancia para la toma de decisiones ya sea de forma inmediata o un control futuro del cultivo (Méndez & Inostroza, 2009).

La toma de decisiones es un proceso de gran importancia dentro de la detección de plagas, porque permite dar una solución idónea a una determinada situación, a partir del monitoreo realizado y evaluaciones realizadas a la planta, seleccionar una de las alternativas de control disponibles y aplicarla al control de la plaga (Ripa, Larral, & Rodríguez, 2008).

Resumiendo, en la **Figura 1** se muestra los pasos principales para realizar el proceso de diagnóstico de plagas en cultivos de papas.

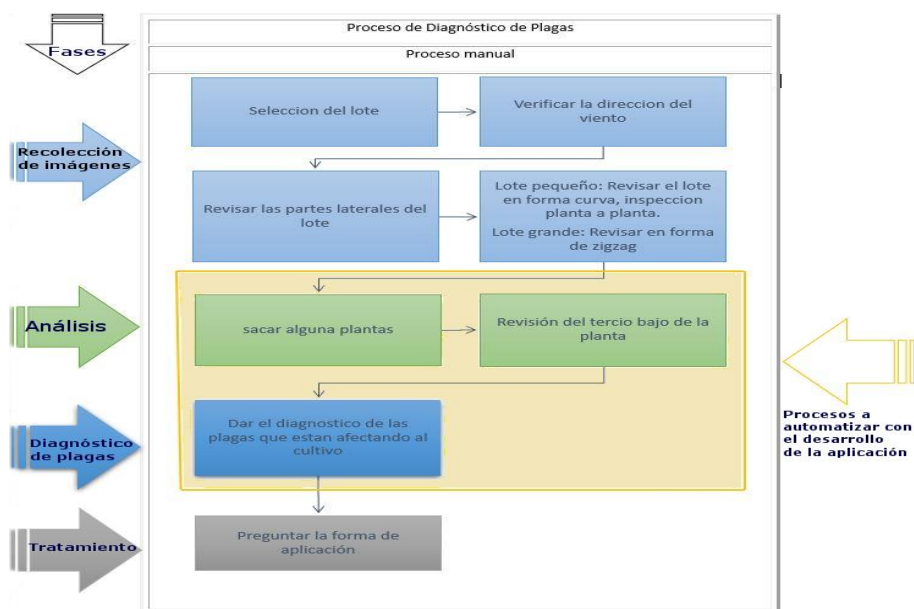


Figura 1. Proceso de Diagnóstico de plagas

d. Grupos de plagas en Cultivos de Papas

Las plagas de los cultivos de papas se pueden dividir en cuatro grupos en términos de la conducta en el campo y de la importancia (Cevallos & Mantilla, 2008).

Plagas constantes: están presentes en la mayoría de cultivos y pueden causar algunas pérdidas económicas o daños cada año, su población varia poco año tras año. El control aplicado generalmente no es suficiente para combatir las grandes cantidades y especies que afectan los cultivos en diferentes etapas de crecimiento (Cevallos & Mantilla, 2008).

Plagas de irrupción: Se pueden encontrar frecuentemente en los cultivos en un nivel bajo, pero son capaces de aumentar masivamente en ciertos periodos y en repuesta a un clima favorable. El aumento puede ocurrir en el cultivo o en huéspedes silvestres, y se producen en el mismo tiempo del año. Es necesario realizar varias inspecciones de los cultivos en momentos críticos o etapas tempranas para poder realizar un control integrado de esta plaga y que no cause un daño severo a la planta (Cevallos & Mantilla, 2008).

Plagas de bajo nivel o auxiliares: Están presentes en baja densidad y en condiciones normales de crecimiento saludable de las plantas, no significan daño la mayor parte de los años, su población no es masivamente grande debido a la acción de insectos enemigos más grandes (Cevallos & Mantilla, 2008).

Vectores: Estos agentes son los que transmiten los virus que se encuentran en el ambiente hacia cultivos o plantas sanas, estos virus producen plagas que afectan a los

cultivos y son difíciles de controlar debido a su agente patógeno¹¹ o por el que fue causado (Cevallos & Mantilla, 2008).

e. Principales plagas que Amenazan los Cultivos de Papas

El instituto Nacional Autónomo de Investigaciones Agropecuarias, en el documento “Quinto curso sobre producción de semilla de papa a partir de cultivo de tejidos, mejoramiento y tecnología de cultivo” plantea que, el cultivo de papa en nuestro país sufre el ataque de diversas plagas durante el transcurso de su ciclo de cultivo, que de no efectuarse un adecuado control producirían severas pérdidas económicas.

Mosca Minadora: Según Bayer CropScience en su sitio web manifiesta que: La mosca minadora es una de las principales plagas que amenazan los cultivos de papas en Sudamérica. Los insectos adultos son de color negro y muestran pequeñas manchas de color amarillo. La hembra ovipone en el envés de las hojas, con la finalidad de proteger los huevos de los rayos solares. Las larvas se alimentan del tejido que se encuentran entre las epidermis del haz y del envés de las hojas, dejando intactas las capas externas. Las hojas afectadas pierden su capacidad fotosintética y pueden llegar a tener la caída de las hojas de la planta totalmente. El control químico para las moscas minadoras se debe realizar con insecticidas selectivos, al observarse los primeros vuelos de adultos, tratando de evitar el crecimiento explosivo de estos insectos (Bayer CropScience, 2019).

¹¹ Que causa o produce una enfermedad

Trips: La plaga Pulguilla Colorada (Trips sp) alrededor de los 50 – 70 días aparece esta plaga, vive preferentemente en el envés de las hojas bajas de la planta, en las que se puede observar insectos de color rojizo o tomate que corresponde a los estados inmaduros, conjuntamente con otros de la misma forma y tamaño, pero de color negro, que son los adultos. El daño que realiza consiste en la succión de los líquidos de las hojas, produciendo una coloración plateada y ataques severos puede provocar defoliación.

Punta Morada: es una enfermedad causada por un Fito plasma (bacteria sin pared) y es transmitida por un insecto, afecta a los cultivos de papas a nivel mundial y recientemente ha sido reportado en Ecuador. Es una enfermedad muy difícil de detectar y controlar, causa grandes pérdidas de rendimiento y calidad a nivel mundial. Organismos Técnicos de territorio como el MAG, AGROCALIDAD e INIAP, así como agricultores y personas vinculadas con el cultivo de papa en el Ecuador trabajan en formas de identificación y alternativas para el manejo de la enfermedad de la punta morada (Instituto Nacional de Investigaciones Agropecuarias, 2018).

f. Instituciones orientadas a la investigación y regulación del sector agropecuario

- **MAG**

El Ministerio de Agricultura y Ganadería (MAG) es una institución del sector agropecuario que se encarga de la agricultura comercial y agricultura familiar campesina, con prioridad en los servicios de asociatividad, comercialización en innovación, para mejorar las condiciones de vida de la población, garantizando la soberanía alimentaria

en el país. La entidad también es encargada de articular los servicios financieros y no financieros, facilitando el desarrollo de los mercados de servicios no financieros, ofreciendo sus servicios de calidad para consolidar un sistema agroalimentario que sea sostenible y sustentable para las personas vinculadas, a nivel social, económico y ambiental (Ministerio de Agricultura y Ganadería, 2019).

El MAG incentiva el cultivo de papa en las comunidades y organizaciones para incrementar la productividad y la economía de las familias, brinda semilla certificada para obtener un rendimiento alto con menor incidencia de plagas y enfermedades y asistencia técnica para el cuidado del cultivo de papa (Ministerio de Agricultura y Ganadería, 2019).

- **INIAP**

El Instituto Nacional de Investigaciones Agropecuarias (INIAP) es una institución adscrita al ministerio de Agricultura y Ganadería, fundado el 11 de julio de 1959, cuyos objetivos primordiales son promover la investigación científica, la generación, innovación, validación y difusión de tecnologías para mejoras en el sector agropecuario y de producción forestal. El INIAP cuenta con 7 estaciones experimentales distribuidas en zonas agroecológicas a nivel nacional, desde donde ejecuta todos sus procesos de investigación, desarrollo e innovación tecnología a nivel de todo el territorio nacional. También cuenta con 13 unidades de desarrollo tecnológico (UDT), 6 granjas experimentales y un invernado automatizado para producir semilla de calidad para el agricultor (Instituto Nacional de Investigaciones Agropecuarias, 2019).

El Instituto Nacional de Investigaciones Agropecuarias (INIAP), trabaja en la generación de nuevas variedades de semilla de papa que permitan satisfacer la necesidad nutricional y mejorar la producción, además desarrolla tecnologías para el manejo de este cultivo con el fin de que el agricultor obtenga un producto sano, de alto rendimiento, el INIAP también trabaja en investigaciones sobre las apariciones de nuevas plagas y enfermedades (Instituto Nacional de Investigaciones Agropecuarias, 2019).

- **AGROCALIDAD**

Es una institución pública adscrita al Ministerio de Agricultura, Ganadería Acuacultura y Pesca, encargada de la ejecución de procesos y políticas de control y regulación para el mejoramiento de la sanidad vegetal, sanidad animal e inocuidad alimenticia mediante las facultades de Autoridad Fitozoo-sanitaria¹² Nacional que posee (Food and Agriculture Organization of the United Nations, 2017).

La misión principal de AGROCALIDAD es encargarse del control de sanidad en el sector agropecuario, impulsar la productividad y competitividad para el desarrollo del sector y mejorar la calidad de vida de productos agropecuarios mediante la ejecución de planes, procesos y proyectos de bienestar animal y vegetal, además de la inocuidad de alimentos, todo con el fin de mejorar la calidad de la producción en los sectores agropecuarios del país (Agencia de Regulación y Control Fito y Zoosanitario, 2019).

Con base en los argumentos citados se puede decir que Agrocalidad es una institución que trabaja con procesos relacionados con el cultivo, cosecha, empaque y transporte de

¹² Que está destinada a prevenir, evitar o combatir cualquier tipo de plaga o enfermedad

papa para consumo y para procesamiento, así como la seguridad de los trabajadores y el cuidado del ambiente. Brinda apoyo para los pequeños, medianos y grandes productores de papa del país que se dediquen a la actividad de cultivo de papas.

2.3.2. Caracterización tecnológica de los métodos y técnicas de diagnóstico de plagas en cultivos de papas

a. Método de detección sintomatológica visual

El método de detección sintomatológica visual consiste en realizar monitoreos constantes al cultivo, y conocer por observación o experimentación las plagas que están afectando al cultivo de papa, las actividades de diagnóstico, en muchos casos suelen ser muy complejas y requieren de una gran experiencia de parte de agricultores o los técnicos agrícolas, así como de alguna formación científica deseable. La experiencia y el conocimiento sobre el cultivo son muy importantes, ya que, si se conoce todo el desarrollo del cultivo, se conocen cuales aspectos bióticos y abióticos interactúan con el cultivo, haciendo muchas veces que el diagnóstico visual sea más sencillo (UNIVERSIDAD DE EL SALVADOR, 2001)

El método de detección sintomatológica visual ha ido evolucionando con el paso de los años con el fin de ser un sistema de protección para los cultivos de papa orientado a mantener las plagas en niveles que no causen daño económico mediante el uso preferencial de componentes naturales o sus derivaciones, que resulten adversos al desarrollo de las plagas en los cultivo, El método tradicional ha evolucionada mediante el desarrollo de guías fotográficas con imágenes de las plagas, de esta manera el

especialista puede comparar las imágenes de la guía, con las halladas en los cultivos, y dar un diagnóstico correcto (Cisneros, 2010).

b. Tecnologías aplicadas a los métodos de diagnóstico de plagas o enfermedades

El método tradicional de detección sintomatológica visual ha sido complementado con la aparición de algoritmos y diferentes técnicas de inteligencia artificial. La tecnología aplicada al campo de la agricultura genera múltiples ventajas como el tratamiento de la información de varios procesos manuales, precisión en la toma de datos y muestras, preservando los recursos que se aplican en una forma tradicional. La producción de un cultivo depende en gran medida de factores como la época de siembra, la aplicación de semilla certificada, el tipo de suelo, la disponibilidad de recursos como agua y alimento de la planta, así como también el control adecuado de plagas y enfermedades en el cultivo y más variables relacionadas con el buen manejo del cultivo por parte de agrónomos y agricultores (Andrade Vera, 2018).

Una de las ramas de la Inteligencia Artificial es el machine learning o aprendizaje automático que permite automatizar operaciones manuales. Para una aplicación de machine learning es necesario disponer de una gran cantidad de información, muestras relevantes o de gran calidad sobre un tema en específico o un determinado objetivo. Es decir, el sistema generaliza comportamientos o decisiones a partir de la información y las muestras obtenidas; de otra manera se traduce como un programa que permite una o varias entradas y posterior a un análisis se obtiene una o varias salidas, un ejemplo práctico está en la detección de enfermedades y plagas en cultivos, a través de la entrada

de imágenes de las plantas mediante la cámara, se puede dar un diagnóstico de la plaga que está amenazando al cultivo (Andrade Vera, 2018).

El uso de técnicas de reconocimiento proporciona facilidades para realizar el análisis de plagas en cultivos extensos de manera rápida y con un diagnóstico acertado, además de muchas más ventajas que se ven reflejadas en la disminución de tiempo aplicado para la detección de plagas tradicional mediante una revisión de todo el cultivo, además de mejorar la calidad de vida de productores y obtener mejor producción y por ende mayores ingresos (Cáceres, Amaya, & Ramos, 2015).

La detección de plagas en cultivos mediante procesamiento de imágenes es un proceso eficiente y con ventajas para agrónomos y agricultores, de forma especial para realizar un monitoreo en grandes extensiones de cultivo, pudiendo obtener un diagnóstico inmediato y aumentar la probabilidad de que la plaga detectada de forma tradicional sea correcta y poder aplicar un control integrado a tiempo (Cáceres, Amaya, & Ramos, 2015).

A continuación, se detallan algunos proyectos relacionados

- **Nuru, una aplicación móvil de PlantVillage para detectar enfermedades de los cultivos en África**

Desde el punto de vista de (Hsiao, 2018), “Para los propietarios de granjas en los países en desarrollo, tener acceso a la educación y las herramientas adecuadas para ayudar en la producción de cultivos es importante para mejorar sus medios de vida. PlantVillage, un grupo de investigación de la Universidad Estatal de Pensilvania,

aprovecha el aprendizaje automático en áreas de conectividad limitada para diagnosticar enfermedades de las plantas y asesorar a los agricultores sobre cómo tratar sus cultivos”.

Actualmente, el grupo de PlantVillage está probando su aplicación móvil, Nuru (“luz” en swahili), para identificar enfermedades y recibir consejos acerca de las mejores soluciones en los cultivos de mandioca, que alimentan a millones de personas en África oriental.

En la **Figura 2** se visualiza el uso de la aplicación móvil desarrollada para detectar enfermedades en la mandioca



Figura 2. Aplicación móvil para detectar enfermedades en la mandioca
Fuente: (Hsiao, 2018)

- **Diseño y Desarrollo de un sistema prototipo de diagnóstico de afecciones en plantas de cítricos utilizando procesamiento de imágenes y aprendizaje profundo**

En la investigación realizada por (VELASCO ZAPATA, 2019) “El sistema consiste en el diseño y desarrollo de un sistema prototipo de diagnóstico de afecciones en plantas de cítricos a partir de fotografías en el rango visible de las hojas de dichas plantas. La construcción de este sistema se basa en los campos del aprendizaje profundo y el procesamiento de imágenes: utilizando un modelo de red neuronal convolucional, para cuyo entrenamiento se aplican transferencia de aprendizaje y aumentado de datos, entre otras técnicas, se construye una aplicación móvil para dispositivos Android que permite la captura de fotografías con la cámara del dispositivo y la obtención de un diagnóstico sobre estas, con base en 7 categorías de afecciones distintas.”

La aplicación móvil permite además el almacenamiento de la información de marca de tiempo, geolocalización y porcentajes de pertenencia a cada categoría para los diagnósticos obtenidos, en un sistema de base de datos en la nube (VELASCO ZAPATA, 2019).

b.1. Tecnologías de Inteligencia artificial aplicadas a la detección de plagas en cultivos

- **Aprendizaje automático**

El Aprendizaje Automático es una rama de la Inteligencia Artificial que se encarga del diseño y desarrollo de algoritmos que permiten a una computadora optimizar un procedimiento automáticamente a través de la práctica. Es por ello que se le da el nombre de Aprendizaje Automático o Aprendizaje de Máquinas, ya que un dispositivo es capaz

de aprender por sí solo por medio de datos empíricos¹³. De forma más concreta, un algoritmo de Aprendizaje Automático es capaz de obtener características y patrones comunes de un conjunto de datos (ejemplos o datos de entrenamiento), y aplicarlas a nuevos conjuntos de datos (datos de pruebas). Esta disciplina se utiliza en diferentes campos, tales como diagnósticos médicos, estudio de mercados, robótica, banca, reconocimiento del habla y lenguaje escrito, descubrimiento de patrones en fotografías, clasificación de secuencias de ADN, o marketing (Nombela Escobar, 2011).

Tipos de aprendizaje:

Aprendizaje supervisado

Los datos se encuentran etiquetados, es decir que se le proporcionan atributos adicionales para que pueda predecir de mejor manera. En esta categoría se encuentran los algoritmos de regresión y clasificación, donde las muestras están etiquetadas y pertenecen a una clase que va ser la salida del algoritmo supervisado.

Aprendizaje no supervisado

Los datos no presentan etiquetas que permitan saber a qué clase pertenece, en este tipo de aprendizaje los datos encuentran características en común y van formando agrupaciones o clustering con elementos que presentan características similares (Gago Utrera, 2017).

¹³ Son datos que se basan en la experiencia y en la observación de los hechos.

- **Redes neuronales Artificiales (Aprendizaje automático)**

Las Redes Neuronales Artificiales son sistemas de procesamiento cuyo trabajo supone el de un pequeño cerebro real. Tienen la capacidad de procesar grandes cantidades de información, aun cuando ésta sea arbitraria y difusa. Pueden aprender y memorizar información muy variada y precisa y, desde luego, hacer predicciones a partir de los datos con los que ha sido entrenada (Vásquez, 2014).

En la **Figura 3** se muestran los elementos que conforman una red neuronal.

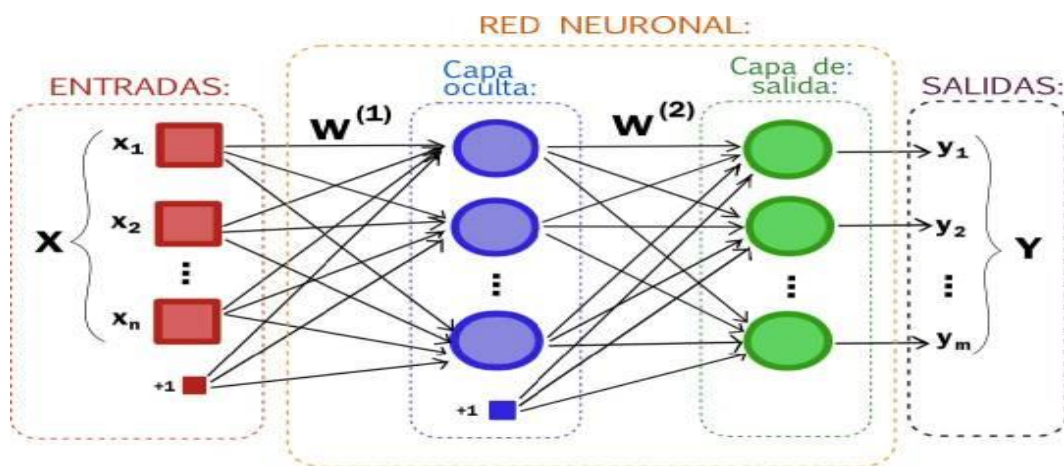


Figura 3. Elementos de una red neuronal
Fuente: (Amorin, 2017)

- **Redes Feedforward**

Las redes feedforward se caracterizan por sus interconexiones que hacen fluir la información en una sola dirección, para que la información no vuelva hacia capas anteriores antes de formarse la respuesta de salida. Este tipo de redes tienen 3 capas de red neuronal, están la capa de entrada donde se recibe toda la información para el entrenamiento de la red neuronal, la capa intermedia donde se realiza todo el proceso

(detección, clasificación) y una capa de salida. El trabajo de la red es integrar los datos a través de la capa de entrada y en la capa oculta recibe el valor de los pesos de las entradas conectadas (Vásquez, 2014).

- **Aprendizaje profundo**

“El Deep Learning o aprendizaje profundo es una colección de técnicas de aprendizaje automático que se utilizan para aprender características que permiten diferenciar objetos de otros, los algoritmos de aprendizaje profundo se basan en redes neuronales artificiales y se utiliza principalmente en temas como el reconocimiento de voz, la visión artificial o el procesamiento del lenguaje natural, todos estos temas son de gran aplicabilidad dentro del mundo real. Las técnicas de aprendizaje profundo permiten expresar representaciones muy complicadas a partir de entradas más simples” (Torres J. , 2016).

El aprendizaje profundo propone que los ordenadores aprendan de la experiencia, y puedan resolver problemas como reconocer palabras o rostros de personas en imágenes, este tipo de aprendizaje automático se caracteriza además de su profundidad por su capacidad para aprender las representaciones de los datos. El rendimiento de los algoritmos de inteligencia artificial depende en gran medida de la cantidad y representación de datos que reciben y muchas veces, es difícil recolectar y seleccionar las características que son las más relevantes, una solución a este problema es utilizar el aprendizaje automático para describir no solo el mapeo de la representación de salida, sino también la representan en sí misma, lo que se conoce como aprendizaje de representación (González, 2018).

Los algoritmos de aprendizaje profundo separan los factores de variación de los datos observados y gracias a ello, son capaces de extraer características abstractas únicas y de alto nivel de los datos proporcionados, con las que pueden abordar eficientemente el problema objetivo. Computacionalmente las redes profundas son modelos complejos obtenidos por la interconexión de una gran cantidad de unidades simples (neuronas artificiales). El poder de estas arquitecturas radica de nuevo, en que muchas operaciones entre elementos simples permiten resolver problemas complejos (González, 2018).

Algunos modelos de aprendizaje profundo son las redes feedforward o perceptrones, que son la base del resto de arquitecturas como redes neuronales convolucionales utilizados principalmente para clasificación y detección de patrones y redes recurrentes. En la última década las técnicas de aprendizaje profundo han evolucionado en el campo de la inteligencia artificial, aportando resultados mayores a los obtenidos aplicando técnicas de aprendizaje automático o machine learning (González, 2018).

Tipos de aprendizaje profundo

- **Redes neuronales convolucionales**

Las redes convolucionales tienen una forma similar a las redes feedforward, la diferencia radica en la introducción de capas convolucionales cuyas neuronas no están completamente conectadas. Cada neurona de las capas puede recibir conexiones entrantes solo de algunas o todas las neuronas de la capa anterior, reduciendo significativamente el número de operaciones a ejecutar. De esta manera las redes convolucionales dividen la información en partes pequeñas para combinar esta

información en las capas más profundas de la red y obtener características comunes rápidamente (González, 2018, págs. 26-29).

En la **Figura 4** se muestra la estructura de una red neuronal convolucional.

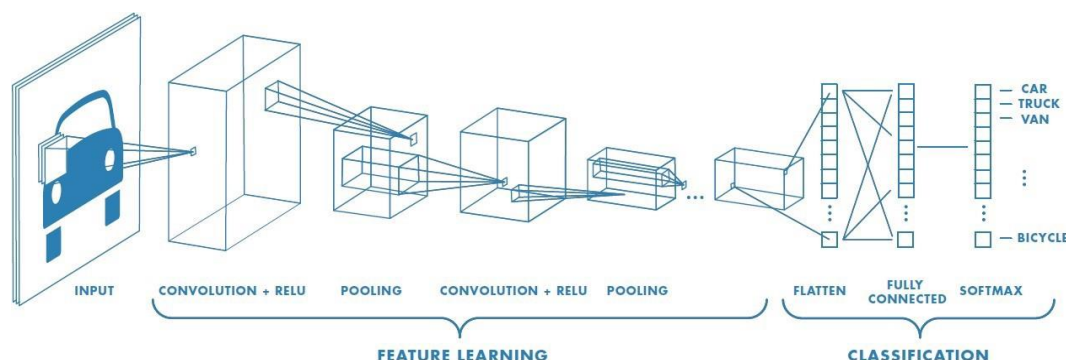


Figura 4. Red Neuronal Convolutiva
Fuente: Recuperado de (Sumit, 2018)

Según Ana González manifiesta que “Por ejemplo, en el caso del tratamiento de una imagen: las primeras capas de la red buscarían contornos, las siguientes tratarían de detectar formas a partir de los contornos, las próximas presentarían atención a la posición de las formas, etc. Por último, en las capas finales del modelo, se combinarían todos los patrones descubiertos para conseguir una predicción final de la suma de todos ellos. Así es como las redes convolucionales consiguen modelar una gran cantidad de datos: dividiendo el problema en partes para conseguir predicciones más sencillas y precisas” (González, 2018, págs. 26-29).

Una red convolucional está formada por las siguientes capas:

Capa convolucional: es una capa oculta donde las neuronas se definen con atributos como los pesos para realizar una convolución con los datos, pueden ser imágenes

permitiendo detectar características comunes, una variable bias (b) y una función de activación para definir la salida de la neurona (Pérez, 2016).

Capa de pooling o submuestreo: es una capa que se conecta entre la salida lineal de la capa convolucional y la función de activación, permitiendo reducir el coste computacional de las siguientes convoluciones, número de parámetros y en ocasiones para controlar el sobre aprendizaje que se puede dar en la red neuronal.

Existen varios tipos de pooling como max pooling y average-pooling, cuando se utiliza max pooling se elige el máximo valor entre un conjunto de píxeles, mientras que en el average pooling se calcula un promedio (Pérez, 2016).

Capas Ocultas: Son capas que reciben el nombre de capa flatten, estas capas son las únicas totalmente conectadas en la arquitectura de red neuronal. Tienen características similares a las redes feedforward, se diferencian que, en el caso de capas ocultas, la primera capa se conecta a la salida de las capas de convolución y capas pooling, con la función de convertir la matriz de datos (imágenes) en un vector plano

- **Entrenamiento de redes neuronales**

Para el entrenamiento de un modelo de aprendizaje profundo se puede llevar a cabo de dos maneras:

Entrenamiento desde cero:

MathWorks manifiesta que “Para entrenar una red profunda desde cero, se recopila un conjunto de datos etiquetados muy amplio y se diseña una arquitectura de red que aprenda las características y el modelo. Esto resulta útil para las aplicaciones nuevas o

las aplicaciones que tendrán un número muy elevado de categorías de salida. Este enfoque es menos común porque, debido a la gran cantidad de datos y a la velocidad de aprendizaje, se suele tardar días o semanas en entrenar estas redes” (MathWorks, 2019).

Transferencia del aprendizaje

Este enfoque es el de los más utilizados a la hora de desarrollar una nueva aplicación, ajustando detalladamente un modelo ya entrenado. Se selecciona un modelo de red neuronal entrenado y se ajusta al nuevo conjunto de datos, modificando el número de salidas. Una de las principales ventajas es, que ya no se necesitan grandes conjuntos de datos, reduciendo así de forma considerable, el tiempo para el proceso de aprendizaje a minutos o horas (MathWorks, 2019).

En la tabla 1 se observa algunos modelos de procesamiento de imágenes pre entrenadas disponibles para utilizar con el enfoque de transferencia de aprendizaje.

Tabla 1

Modelos pre-entrenados para aprendizaje profundo de TensorFlow

| Nombre del modelo | Velocidad | Precision | Salidas |
|----------------------------|-----------|-----------|---------|
| ssd_mobilenet_v1_coco | 30 | 21 | Boxes |
| ssd_resnet_50_fpn_coco | 76 | 35 | Boxes |
| ssd_inception_v2_coco | 42 | 24 | Boxes |
| faster_rcnn_resnet101_coco | 106 | | Boxes |

Fuente: (TensorFlow, 2019)

- **Usos del aprendizaje profundo**

“Entre los principales usos del Aprendizaje Profundo en la actualidad se encuentra identificar marcas y logotipos de empresas en fotos publicadas en las redes sociales. Monitorización en tiempo real de reacciones en canales online durante el lanzamiento de productos, Identificación de clientes potenciales, detección de fraudes, recomendación a

clientes, gestión de relaciones con los clientes, entre otros. Mejor comprensión de enfermedades y terapias genéticas, análisis de imágenes médicas, como radiografías y resonancias magnéticas, aumentando la precisión de diagnóstico, en un menor tiempo y con un menor coste que los métodos tradicionales, identificación en textos de sentimientos positivos y negativos, temas, palabras claves. Localización de caras e identificación de emociones faciales, así como su aplicación en la agricultura en los últimos años” (Ávila, Eduardo, Orozco, Sánchez, & Hidalgo, 2018) .

- **Tareas del aprendizaje profundo**

Clasificación de Imágenes

La clasificación es un arreglo sistemático en grupos y categorías basado en sus características. La clasificación de imágenes se creó para la disminución de la brecha entre la visión por computador y la visión humana entrenando a la computadora con los datos. Los métodos convencionales utilizados para la clasificación de imágenes son parte del campo de la inteligencia artificial (IA) formalmente llamado como aprendizaje automático. El aprendizaje automático consiste en un módulo de extracción de características que extrae las características importantes como bordes, texturas, etc. y un módulo de clasificación que clasifica en función de las características extraídas. La principal limitación del aprendizaje automático es que, separando, sólo puede extraer cierto conjunto de características de las imágenes e incapaz de extraer características diferenciadoras del conjunto de datos. Esta desventaja se rectifica mediante el uso del Deep Learning o aprendizaje profundo. El aprendizaje profundo (DL) es un subcampo del aprendizaje automático, capaz de aprender a través de su propio método de

computación. Un aprendizaje profundo introduce un modelo para desglosar la información de forma persistente con una estructura homogénea como la forma en que un humano haría las determinaciones. Para lograr esto, el aprendizaje profundo utiliza una estructura en capas de varios algoritmos expresados como un sistema neural artificial (ANN). La arquitectura de una red neuronal artificial (RNA) se simula con la ayuda de la red neuronal biológica del cerebro humano. Esto hace que el aprendizaje profundo sea más capaz que el aprendizaje estándar de la máquina (M Manoj, Neelima, Harshali, & M Venu, 2018).

En el aprendizaje profundo, consideramos las redes neuronales que identifican una imagen basada en sus características. Esto se logra mediante la construcción de un modelo completo de extracción de características que es capaz de resolver las dificultades debidas a los métodos convencionales. El extractor del modelo integrado debe ser capaz de aprender a extraer con precisión las características del conjunto de imágenes (M Manoj, Neelima, Harshali, & M Venu, 2018).

Detección de objetos

La detección de objetos es el procedimiento para determinar la instancia de la clase a la que pertenece el objeto y estimar la ubicación del objeto mediante la salida del cuadro delimitador alrededor del objeto. Detección de una sola instancia de la imagen se llama detección de objetos de clase única, mientras que la detección de las clases de todos los objetos presentes en el archivo se conoce como detección de objetos de clase múltiple. Diferentes retos como la oclusión parcial o total, diferentes condiciones de iluminación,

posturas, escala, etc. son necesarias para ser manejadas mientras se realiza la detección de objetos (Ajeet, Manjusha, & Siddharth, 2018).

Detección de objetos mediante redes neuronales convolucionales

Las redes neuronales convolucionales CNNs profundas han sido ampliamente utilizadas para la detección de objetos. La CNN es un tipo de red neuronal de retroalimentación y trabaja bajo el principio de compartir el peso. La convolución es una integración que muestra cómo una función se superpone a otra y es una mezcla de dos funciones que se multiplican. La **Figura 5** muestra la arquitectura en capas de redes neuronales convolucionales (CNN) para el objeto detección. La imagen se convoluciona con la función de activación para obtener mapas de características. Para reducir la complejidad espacial de los mapas de características se tratan con capas de agrupamiento para obtener mapas de características abstractos. Este proceso se repite para el número deseado de filtros y, en consecuencia, se crean mapas de características. Eventualmente, estos mapas de características se procesan con capas completamente conectadas para obtener un resultado de reconocimiento de imagen que muestra la puntuación de confianza para la clase pronosticada de etiquetas. Para mejorar la complejidad de la red y reducir el número de parámetros, CNN emplea a diferentes tipos de capas de pooling (Ajeet, Manjusha, & Siddharth, 2018).

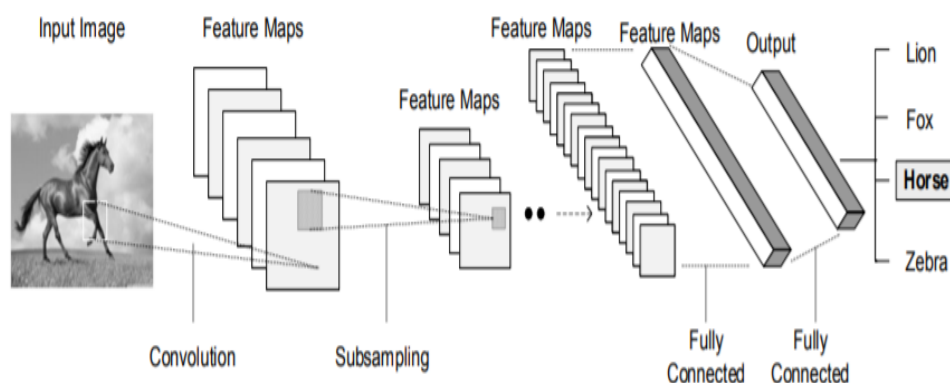


Figura 5. Arquitectura de una red neuronal convolucional

Fuente: (Ajeet, Manjusha, & Siddharth, 2018)

Reconocimiento de voz

El reconocimiento automático de voz es la capacidad de una máquina o computadora para reconocer el contenido de palabras y frases en un lenguaje pronunciado y transformarlas en un formato comprensible para la máquina. El reconocimiento de voz se puede utilizar en muchas aplicaciones. Estas aplicaciones aparecen en: dictado de computadoras en lugar de escribir, naves espaciales cuando las extremidades están ocupadas, ayuda a personas discapacitadas, hogares inteligentes y muchos otros (Bou, Shahin, Attili, & Shaalan, 2019).

El reconocimiento de voz es una de las áreas más desafiantes en la informática, una gran cantidad de metodología de reconocimiento de patrones trató de resolver el problema con un mayor porcentaje de reconocimiento. Los retos del aprendizaje profundo son hacer que las máquinas inteligentes no sean capaces de sólo oír (habla) y ver (visión), sino también pensar con la mente, es decir, razonar e interactuar sobre una base

compleja y jerárquica de relaciones y fuentes de conocimiento que comprenden un gran número de entidades y conceptos semánticos en el mundo real basados en parte en datos multisensoriales¹⁴ del usuario. habla (audio) e imagen (video) está evolucionando hacia una nueva frontera de aprendizaje profundo, comenzando por una mezcla de investigación de comunidades de procesamiento del habla y del lenguaje hablado, procesamiento del lenguaje natural, visión por computador, aprendizaje por computadora (Rubi, 2015).

b.2. Herramientas para el desarrollo de sistema software de aprendizaje profundo

- **Google Colaboratory**

Google Colaboratory es un servicio gratuito en la nube que proporciona herramientas preinstaladas como TensorFlow y GPUs¹⁵ gratuitos para ejecutar los modelos realizados. Actualmente, Google Colaboratory proporciona los servicios de una GPU tesla K80 de forma gratuita. El problema que se presenta es que solo se puede hacer uso de estos servicios informáticos durante un tiempo máximo de 12 horas como si terminara el tiempo de una sesión. Básicamente cuando se entrenan modelos en Google Colaboratory, el entrenamiento se basa en un máquina virtual basada en GPU donde se le da un máximo de 12 horas a las vez, después de este tiempo se pierde acceso a esta instancia de máquina virtual en particular, Es decir se pierde acceso a la información subida a la nube, los modelos, conjunto de datos que no se guarden en Google o se descarguen antes de

¹⁴ Datos obtenidos a través del uso de todos los sentidos como vista, oído, etc.

¹⁵ Unidades de procesamiento gráfico

que se pierda este periodo de tiempo, así que asegúrese de guardar sus modelos entrenados de lo contrario tendrá que empezar nuevamente . Después de 12 horas, se le asigna una máquina virtual diferente (de forma gratuita, por supuesto) y el ciclo se repite (Linga, 2018).

Tiene como grandes ventajas:

- Posibilidad de activar una GPU
- Podemos compartir el código fácilmente
- Está basado en jupyter notebook y nos resultará un entorno ya conocido
- Podemos crear libros en Python 2 ó en 3
- Tiene preinstaladas las librerías comunes usadas en data science y la posibilidad de instalar otras que necesitemos
- Al enlazar con nuestra cuenta de Google Drive, podemos leer desde ahí archivos csv¹⁶ de entrada o guardar imágenes de salida, etc (Aprende Machine Learning, 2019).
- **TensorFlow**

Son librerías de código abierto, donde este marco de trabajo opera sobre lo que constituye aprendizaje profundo. Este tipo de algoritmos, se han utilizado por varios años en muchos productos y áreas en Google, como lo son búsqueda, traducción, publicidad, visión artificial y reconocimiento de voz e imágenes.

TensorFlow es un sistema de segunda generación, para implementar y desplegar redes neuronales. Fue lanzado al público como un marco de código abierto con Apache

¹⁶ comma-separated values en español valores separados por comas

2.0, en noviembre del 2015, donde en sus primeros inicios, se dedicó a proyectos internos de Google, pero, gracias a su estabilidad y flexibilidad, combinado con la experiencia de los Ingenieros de Google, que son el motor que da mantenimiento y que continúa desarrollando las librerías, han convertido a TensorFlow en un sistema líder para hacer aprendizaje profundo (Espinoza Mendoza, 2018).

La arquitectura flexible de TensorFlow le permite implementar el cálculo a una o más CPU o GPU en equipos de escritorio, servidores o dispositivos móviles con una sola API¹⁷. TensorFlow fue desarrollado originalmente por investigadores e ingenieros que trabajaban en el equipo de Google Brain Team, dentro del departamento de investigación de Machine Intelligence, con el propósito de llevar a cabo el aprendizaje automático y la investigación de redes neuronales profundas. Sin embargo, el sistema es lo suficientemente general como para ser aplicable a una amplia variedad de otros dominios igualmente (Buhigas, 2018).

Las API de alto nivel de TensorFlow se basan en el estándar Keras API para definir y entrenar redes neuronales. Keras permite la creación rápida de prototipos, la investigación más avanzada y la producción, todo con API fáciles de usar

- **TensorFlow Lite**

TensorFlow Lite es un conjunto de herramientas para ayudar a los desarrolladores a ejecutar modelos de TensorFlow en dispositivos móviles, integrados e Internet de las

¹⁷ Interfaz de programación de aplicaciones

cosas (IoT). Permite la inferencia de aprendizaje automático en el dispositivo con baja latencia y un tamaño binario pequeño (TensorFlow, 2019).

TensorFlow Lite consta de dos componentes principales:

El intérprete TensorFlow Lite, que ejecuta modelos especialmente optimizados en diferentes tipos de hardware, incluidos teléfonos móviles, dispositivos Linux integrados y microcontroladores (TensorFlow, 2019).

El convertidor TensorFlow Lite, que convierte los modelos TensorFlow en una forma eficiente para que los utilice el intérprete, puede introducir optimizaciones para mejorar el tamaño y el rendimiento binarios (TensorFlow, 2019).

Flujo de trabajo de desarrollo

El flujo de trabajo para usar TensorFlow Lite implica los siguientes pasos:

Elige un modelo

Traiga su propio modelo TensorFlow, encuentre un modelo en línea o elija uno de nuestros modelos pre-entrenados como Clasificación de Imágenes, Detección de Objetos, Respuesta Inteligente, Estimación de la Postura, Segmentación para dejar caer o volver a entrenar (TensorFlow, 2019).

Convertir el modelo

Si está utilizando un modelo personalizado, use el convertidor TensorFlow Lite y algunas líneas de Python para convertirlo al formato TensorFlow Lite (TensorFlow, 2019).

Implementar en su dispositivo

Ejecute su modelo en el dispositivo Android con el intérprete TensorFlow Lite, con API en muchos idiomas (TensorFlow, 2019).

Optimiza tu modelo

Utilice nuestro kit de herramientas de optimización de modelos para reducir el tamaño de su modelo y aumentar su eficiencia con un impacto mínimo en la precisión (TensorFlow, 2019).

- **Python**

Python es un lenguaje de programación muy potente y fácil de aprender. Cuenta con estructuras completas de datos eficientes y de alto nivel y su enfoque hacia la programación orientada a objetos. Python presenta una elegante sintaxis y cuenta con un tipado dinámico, además de su naturaleza interpretada, todas estas características hacen de este un lenguaje ideal para desarrollo rápido de variedad de aplicaciones en diferentes áreas y sobre la mayoría de plataformas. El intérprete de Python y la extensa biblioteca estándar están a libre disposición en código fuente y de forma binaria para las principales plataformas desde el sitio web de Python, <https://www.python.org/>, y puede distribuirse libremente (van Rossum, 2017).

Python es uno de los lenguajes más utilizados para configurar Inteligencia Artificial (IA). Su simplicidad y las filosofías DRY (Don't Repeat Yourself) y RAD (Rapid Application Development) en las que se basa lo convierten en un candidato ideal. Puede utilizarse tanto para estructurar datos como para generar algoritmos de IA. Además, dispone de un catálogo de librerías muy extenso que permite hacer realidad cualquier tipo de proyecto. Ningún soporte se resiste a Python, puesto que sirve para trabajar en aplicaciones de todo tipo, sean móviles, web, data science o IA.

- **Android Studio**

Android Studio es un entorno de desarrollo integrado (IDE) que proporciona la interfaz para crear aplicaciones y manejar gran parte de la complicada administración de archivos detrás de la escena. El lenguaje de programación con el que trabaja Android Studio es Java o kotlin. Si es Java, se instalará por separado en la máquina. Android Studio es simplemente donde se escribe, guarda, edita y guarda todos los proyectos y los archivos correspondientes a dichos proyectos, del mismo modo Android Studio le dará acceso al kit de desarrollo de software (SDK) de Android. (Mullis, 2017).

Se necesita java para escribir los programas, y se necesita el SDK de Android para poder ejecutar la aplicación en un dispositivo Android y Android Studio tiene la tarea de ponerlo todo junto para el usuario. Al mismo tiempo, Android Studio también le permite ejecutar su código, ya sea a través de un emulador nativo, emulador externo instalada o mediante una pieza de hardware conectada a su máquina. Entonces, también podrá depurar el programa mientras se ejecuta y obtener comentarios que explican los bloqueos, etc. (Mullis, 2017).

- **Java**

JAVA es un lenguaje de programación que utiliza un compilador, para traducir del código fuente al código ejecutable, aunque también tiene la característica de ser un lenguaje interpretado. El compilador de JAVA genera a partir del código fuente de un programa un código intermedio llamado bytecode, el cual es independiente de la plataforma en que se trabaje y este bytecode solo se puede ejecutar en la máquina virtual de java conocida como JVM, la cual es una estructura idealizada de java y en la mayoría de los casos es utilizada para implementar software en vez de hardware. (Arenas Hernández).

Java es un estándar para el desarrollo y distribución de aplicaciones móviles y embebidas, juegos y aplicaciones web, con más de 9 millones de desarrolladores en todo el mundo, este lenguaje permite desarrollar y utilizar aplicaciones y servicios de forma eficaz. (Java, 2016).

c. Metodología Mobile-D

c.1. Introducción

El sector de las aplicaciones móviles ha evolucionado de manera muy rápida en pocos años. Esto ha permitido crear metodologías de desarrollo de proyectos enfocados a la parte móvil que sean rápidos y eficaces.

Mobile-D es una metodología ágil de desarrollo de software, enfocado a aplicaciones móviles, con la finalidad de obtener pequeños ciclos de desarrollo rápido y fácil con grupos de trabajo pequeños.

La metodología Mobile-D se creó en el año 2004, por un grupo de investigadores de nacionalidad finlandés. Desde su creación hasta la actualidad ha ido tomando un mayor grado de importancia, al ser una metodología que permite crear proyectos de gran calidad, debido a sus métodos y técnicas que son muy prácticos y fáciles de utilizar (Álban & Bassante, 2015).

c.2. Principios

Mobile-D está basada en 3 principales metodologías de desarrollo de software: Extreme Programming para las prácticas de desarrollo, Crystal Methodologies para escalar métodos y Rational Unified Process como base del diseño del ciclo de vida (Álban & Bassante, 2015).

c.3. Fases del Mobile-D

Mobile-D es una metodología ágil de desarrollo de software. El ciclo de cualquier proyecto en esta metodología es rápido y fácil. Consta de cinco fases: exploración, iniciación, producción, estabilización y pruebas del sistema (Balaguera, 2013).

a. Fase de exploración

En esta fase, el grupo encargado del desarrollo debe generar un plan y determinar las características esenciales del proyecto. Para ello se debe determinar los siguientes puntos: establecimiento de los actores, definición del alcance y el establecimiento de proyectos. También el cliente toma un papel importante en el proceso de desarrollo, planificación inicial del proyecto, la toma de requisitos y el establecimiento de los procesos (Balaguera, 2013).

b. Fase de inicialización

En esta fase el equipo de trabajo debe preparar los recursos necesarios con la finalidad asegurar el éxito de las siguientes fases de desarrollo. También se debe preparar todos los recursos tecnológicos, físicos y de comunicaciones para las actividades de producción. Esta fase incluye cuatro instancias: la puesta en marcha del proyecto, la planificación inicial, el día de prueba y el día de salida (Balaguera, 2013).

c. Fase de producción

En esta fase se realiza la implementación del proyecto con duración de 3 días (planificación, trabajo y liberación), esto es repetitivo hasta culminar con toda la funcionalidad del sistema. Como primer paso es necesario planificar la iteración en términos de los requisitos y todas las actividades a realizar. De igual manera se elabora todas las pruebas pertinentes para esta iteración. Estas tareas son realizadas durante el día de trabajo. El ultimo día es ocupado para realizar las diferentes pruebas de aceptación (Balaguera, 2013).

d. Fase de estabilización

Esta fase permite realizar las ultimas tareas de integración del sistema, con la única finalidad de que el sistema completo funcione correctamente. En esta fase también se realiza la documentación del sistema (Balaguera, 2013).

e. Fase de pruebas del sistema

Esta fase permite buscar la disponibilidad de una versión estable y funcional del sistema. Este producto terminado se prueba con los requisitos del cliente y se eliminarán todos los defectos y errores encontrados (Balaguera, 2013).

c.4. Ventajas

- Se realiza de manera rápida y efectiva el proyecto.
- El tiempo que se demora en realizar el proyecto es corto
- No se requieren de muchas personas para el proyecto.
- Asegura la calidad del producto desarrollado.

c.5. Desventajas

- Se necesita los requisitos bien detallados.
- Solo está enfocado en aplicaciones móviles.
- Enfocado solo para proyectos pequeños.

2.4. Antecedentes Contextuales

La parroquia de Toacaso se encuentra ubicado en la provincia de Cotopaxi específicamente en la parte noroccidental del cantón Latacunga en las faldas de los Ilinizas (Gobierno Autónomo Descentralizado Parroquia Rural de Toacaso, 2019).

La población se dedica al comercio de los siguientes productos: habas, abarrotes, papas, harinas, venta de animales entre otros (Gobierno Autónomo Descentralizado Parroquia Rural de Toacaso, 2019).

De acuerdo a una entrevista realizada a los técnicos especializados del Ministerio de Agricultura y Ganadería (MAG) a cargo de la zona de influencia en el proyecto de investigación, se siembran anualmente alrededor de 4.500 a 5000 hectáreas de papas. En la parroquia de Toacaso el sector con más cultivos de papas es la Libertad en donde se ha determinado la presencia de plagas como: mosca minadora y psilido de la punta morada, siendo las principales amenazas en los cultivos de papas.

Para corroborar la existencia del problema orientado al diagnóstico de plagas en cultivos de papas en el sector la Libertad de la parroquia de Toacaso, se elaboró una encuesta a los moradores del sector y al ingeniero agrónomo encargado de dicho sector, de las cuales se obtuvieron los siguientes resultados:

1. ¿Con que frecuencia existe la presencia de plagas en su cultivo de papas?

Interpretación: Los resultados de esta pregunta reflejan que el 100% considera que “siempre” existe la presencia de plagas en cultivos de papas.

2. ¿Cómo usted determina la presencia de plagas en su cultivo de papas?

Interpretación: Del todas de personas a las que se aplicó la encuesta el 84% considera que la presencia de plagas se determina mediante detección visual, el 8% mediante guías fotográficas y el 8% mediante la experiencia del agricultor, lo cual demuestra que la principal forma de determinar la presencia de plagas es mediante detección visual.

3. ¿Cuál es el porcentaje de acierto que se tiene al realizar el diagnóstico de plagas en cultivos de papas?

Interpretación: El 70% de la muestra considera que el porcentaje de acierto que se tiene al realizar el diagnóstico de plagas en cultivos de papas es bajo y el 30% considera que el porcentaje de acierto es medio, demostrando que el porcentaje de acierto es bajo al momento de dar un diagnóstico de plagas del cultivo.

4. ¿Existen pérdidas de producción por presencia de plagas en cultivos de papas?

Interpretación: De acuerdo con los datos el 100% de la muestra expresa que, si existen pérdidas de producción por presencia de plagas en cultivos de papas.

5. ¿Conoce usted de alguna aplicación que haga uso de tecnologías para el proceso de diagnóstico de plagas en cultivos de papas?

Interpretación: De todas las personas encuestadas el 100% considera que no conocen alguna aplicación que haga uso de tecnologías para el proceso de diagnóstico de plagas en cultivos de papas, esto indica que las personas desconocen totalmente de tecnologías aplicadas en este campo.

6. ¿Si se desarrolla una aplicación para el diagnóstico de plagas en cultivos de papas usted estaría dispuesto a utilizarla?

Interpretación: El enfoque de la pregunta 8 es para determinar la aceptabilidad de los usuarios hacia la propuesta planteada logrando un 100% de respuesta positivas lo que demuestra que en general la población se ve atraída a una solución de este tipo.

7. ¿Considera usted que mediante el uso de tecnologías el proceso de diagnóstico de plagas en cultivos de papas sería más efectivo?

Interpretación: Los resultados de la pregunta reflejan que un 100% considera que el uso de tecnología si sería más efectivo para el control de plagas, determinando que la población usaría tecnología para mejorar el diagnóstico de plagas en sus cultivos.

En base a la encuesta realizada se pudo ver que las personas encuestadas mencionan que siempre existe la presencia de plagas en los cultivos de papas y el problema radica en que el porcentaje de acierto que se tiene al realizar un diagnóstico es bajo, razón por la cual existen grandes pérdidas de producción por la presencia de plagas.

Las personas encuestadas también indican que, mediante el uso de tecnologías, el proceso de diagnóstico de plagas en cultivos de papas si sería más efectivo, por lo tanto, la solución del problema sería el desarrollada la aplicación para el diagnóstico de plagas, aplicación que las personas estarían dispuestas a utilizarla.

CAPÍTULO III

DESARROLLO DEL SISTEMA SOFTWARE DE APRENDIZAJE PROFUNDO QUE OPTIMICE EL DIAGNÓSTICO DE PLAGAS EN HOJAS DE CULTIVOS DE PAPAS.

3.1. Introducción al capítulo

En este capítulo se describe el desarrollo de la propuesta “Sistema software de aprendizaje profundo que optimice el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertad parroquia Toacaso provincia de Cotopaxi”, el sistema software esta dividido en dos instancias.

En primera instancia se inicia con el desarrollo del modelo de red neuronal, tomando en cuenta los principios que existen en el campo del aprendizaje profundo para el desarrollo de dicho modelo. La red neuronal será desarrollada mediante la arquitectura MobileNet, que son redes neuronales convolucionales, eficientes para el desarrollo de aplicaciones móviles. En segunda instancia consiste en el desarrollo de la aplicación móvil siguiendo los procesos de ingeniería en software y mediante el uso de la metodología orientada al desarrollo de aplicaciones móviles llamado Mobile-D. La aplicación móvil utilizara el modelo de red neuronal desarrollada en la primera instancia para el diagnóstico de plagas en hojas de cultivos de papas, también la aplicación contiene información detallada sobre las diferentes plagas que amenazan los cultivos de papas, información sobre los diferentes insecticidas que se pueden aplicar una vez

detectado posible amenaza de plagas e información detallada sobre el proceso de diagnóstico de plagas.

3.2. Modelos de redes neuronales

Un modelo de red neuronal convolucional es un tipo de red neuronal artificial especializado en el tratamiento de imágenes, que procesa sus capas imitando al córtex visual del ojo humano con el objetivo de identificar distintas características en las entradas que en definitiva hacen que puedan identificar objetos en una imagen. Como resultado las redes neuronales convolucionales aprenden a responder a características simples de una imagen (bordes, forma, etc.), hasta características complejas como un rostro una silueta de un animal (Rodríguez J. , 2018).

En la actualidad, existen multitud de arquitecturas de redes neuronales convolucionales disponibles de forma gratuita y sin restricciones de uso que puedan alcanzar rendimiento razonable en tareas de reconocimiento visual.

A continuación, se detalla algunas arquitecturas de redes neuronales convolucionales que son usados en el campo del aprendizaje profundo:

- **Inception-v3**

Arquitectura usada en el aprendizaje profundo para el reconocimiento de imágenes, desarrollado por Google por el año del 2015 el cual se demostró que alcanza una exactitud superior al 78.1% en el conjunto de datos de ImageNet¹⁸. Está basado en redes

¹⁸ Es una gran base de datos visual diseñada para el uso en modelos de redes neuronales

neuronales convolucionales debido a que son capaces de modelar complejas variaciones y comportamientos dando predicciones bastantes precisas (Quiza, 2018).

- **ResNet**

Arquitectura utilizada en el campo del aprendizaje profundo, desarrollado por el investigador científico Kaiming He a finales del año 2015 con el objetivo de desarrollar aplicaciones de visión por computadora. Su mayor impacto se debe a que el paradigma ResNet permitió por primera vez entrenar redes convolucionales con más de 100 capas. La idea se basa en aumentar el número de capas introduciendo una conexión residual (con una capa identidad). Esta capa pasa a la siguiente directamente, mejorando el proceso de aprendizaje (Utrera, 2018).

- **MobileNet**

Arquitectura desarrollada por Google y basada en redes neuronales convolucionales adecuadas para tecnologías móviles, para construir redes ligeras y profundas, diseñado para maximizar la precisión considerando los recursos limitados de un dispositivo móvil. Usada para realizar tareas de aprendizaje profundo enfocadas al desarrollo de aplicaciones móviles como: clasificación de imágenes, detección de objetos y respuesta inteligente (Howard, 2017).

3.2.1. Análisis comparativo entre arquitecturas de redes neuronales convolucionales

Previo al desarrollo del modelo de red neuronal, se realiza un estudio comparativo entre las arquitecturas de redes neuronales convolucionales, con el propósito de

determinar que arquitectura de red neuronal convolucional es la más adecuada para el desarrollo de aplicaciones móviles.

El estudio comparativo consiste en establecer las similitudes y diferencias entre las arquitecturas de redes neuronales convolucionales mencionadas en los apartados anteriores en esta investigación, con el fin de definir la arquitectura adecuada para el desarrollo de la aplicación móvil que permita realizar el proceso de diagnóstico de plagas de manera eficiente. Los criterios que se establecen para comparar están relacionados a: tecnologías adecuadas para dispositivos móviles, redes neuronales livianas, especializadas en el reconocimiento de imágenes y con antecedentes en ambientes agrícolas.

Tabla 2

Criterios para seleccionar el modelo de red neuronal adecuado para el uso en aplicaciones móviles. “Cuadro comparativo de redes neuronales”

| Criterios | Inception-V3 | ResNet | MobileNet |
|--|--------------|--------|-----------|
| Soporte para tecnologías móviles | No | No | Si |
| Redes neuronales livianas | No | Si | Si |
| Reconocimiento de imágenes | Si | Si | Si |
| Antecedentes en ambientes agrícolas | No | No | Si |

De la tabla 9 se puede decir que la arquitectura más adecuada para el desarrollo del modelo de red neuronal convolucional es MobileNet, debido a que cumple con todos los criterios establecidos, permitiendo que la aplicación móvil desarrollada pueda dar un diagnóstico correcto sobre la presencia de plagas en los cultivos de papas.

3.3. Proceso de desarrollo del modelo de red neuronal convolucional para el diagnóstico de plagas en cultivos de papas

En la **Figura 6** se muestran el conjunto de pasos para el desarrollo del modelo de red neuronal del proyecto de investigación.

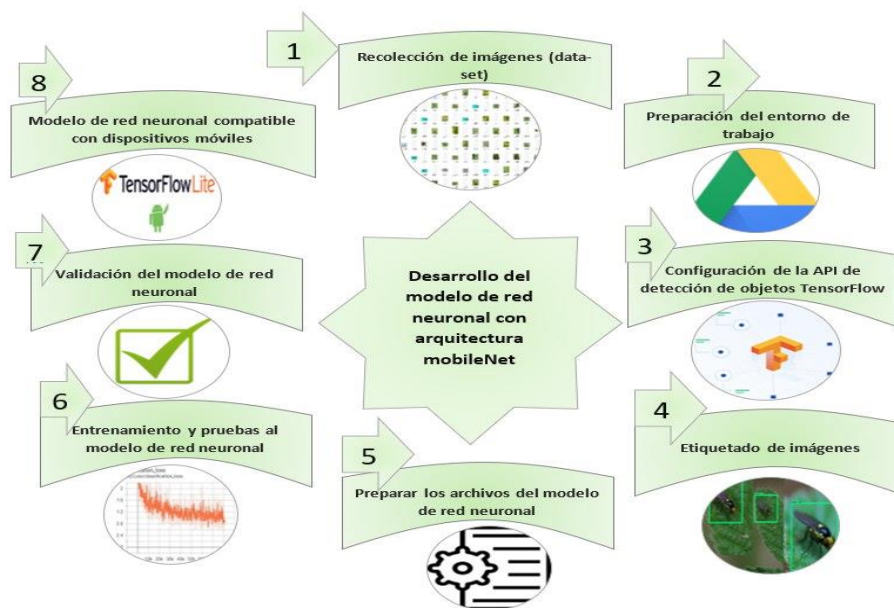


Figura 6. Desarrollo del modelo de red neuronal

3.4. Desarrollo del modelo de red neuronal convolucional

El desarrollo del modelo constituye los pasos mencionados en la **Figura 6**, estos pasos permiten la implementación del modelo para su posterior validación y ejecución en un ambiente controlado, A continuación, se detalla cada uno de ellos.

3.4.1. Recolección de imágenes (dataset)

Con la información obtenida de las principales plagas que existen en los cultivos de papas se procede conjuntamente con el técnico a cargo del sector la Libertad a realizar

la recolección de imágenes de las dos principales plagas, psílido de la punta morada y mosca minadora, para ello se realiza un recorrido a los diferentes lotes de cultivos de papas para la toma de imágenes. Estas imágenes contienen el insecto adulto que amenaza la planta a nivel de las hojas.

En la **Figura 7** se visualiza algunas imágenes del insecto de la punta morada del conjunto de datos obtenido en los diferentes campos con ambientes controlados.

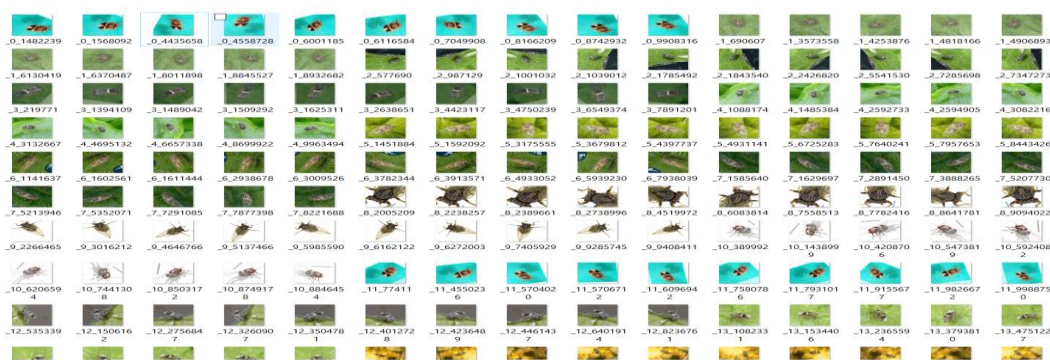


Figura 7. Psílido de la punta morada

En la **Figura 8** se muestra el insecto adulto, técnicamente llamado bactericera cockerelli causante de la plaga de la punta morada.



Figura 8. Bactericera cockerelli

En la **Figura 9** se visualiza algunas imágenes del insecto de la mosca minadora del conjunto de datos obtenido en los diferentes campos con ambientes controlados.



Figura 9. Mosca minadora de la papa

En la **Figura 10** se muestra el insecto adulto, causante de la plaga mosca minadora.



Figura 10. Mosca minadora adulta

En total se han recolectado 20000 imágenes, las cuales serán utilizadas para el entrenamiento y validación del modelo de red neuronal.

3.4.2. Preparación del entorno de trabajo

Para utilizar la API de detección de objetos de TensorFlow se hizo uso de Google Colaboratory que es un entorno de máquinas virtuales basado en el entorno de desarrollo integrado (IDE) Jupyter Notebooks, que no requiere configuración y se ejecuta completamente en la nube. Colaboratory te permite escribir y ejecutar código, guardar y compartir tus análisis y tener acceso a recursos informáticos muy potentes, todo de forma gratuita desde el navegador.

3.4.3. Configuración de la API de detección de objetos de Tensorflow

Para construir el modelo se utiliza la API de detección de objetos de TensorFlow debido a que proporciona diferentes modelos pre-entrenados entre ellos modelos basados en arquitectura MobileNet.

La API de detección de objetos de TensorFlow permite crear modelos precisos de aprendizaje profundo, capaces de identificar, ubicar múltiples objetos en una imagen, en nuestro caso el modelo de red neuronal es capaz de reconocer y localizar en tiempo real la presencia de insectos adultos de la plaga.

Para utilizar la API de detección de objetos de TensorFlow es necesario clonar o descargar desde su página oficial en GitHub.

En la **Figura 11** se muestran los archivos necesarios para utilizar el api de detección de objetos que proporciona TensorFlow.

| | | | |
|----------------------------------|----------------------|-------------|-------|
| anchor_generators | 5/31/2019 5:22 PM | File folder | |
| box_coders | 5/31/2019 5:22 PM | File folder | |
| builders | 5/31/2019 5:22 PM | File folder | |
| core | 5/31/2019 3:05 PM | File folder | |
| data | 5/21/2019 10:43 A... | File folder | |
| data_decoders | 5/31/2019 5:22 PM | File folder | |
| dataset_tools | 5/21/2019 10:43 A... | File folder | |
| dockerfiles | 5/21/2019 10:43 A... | File folder | |
| g3doc | 5/21/2019 10:43 A... | File folder | |
| inference | 5/21/2019 10:43 A... | File folder | |
| legacy | 5/31/2019 5:22 PM | File folder | |
| matchers | 5/31/2019 5:22 PM | File folder | |
| meta_architectures | 5/31/2019 5:22 PM | File folder | |
| metrics | 5/21/2019 10:43 A... | File folder | |
| models | 5/31/2019 5:22 PM | File folder | |
| predictors | 5/31/2019 5:22 PM | File folder | |
| protos | 5/31/2019 3:05 PM | File folder | |
| samples | 5/21/2019 10:43 A... | File folder | |
| tensorflow-master | 6/13/2019 1:22 PM | File folder | |
| test_ckpt | 5/21/2019 10:43 A... | File folder | |
| test_data | 5/21/2019 10:43 A... | File folder | |
| test_images | 5/21/2019 10:43 A... | File folder | |
| utils | 5/31/2019 3:05 PM | File folder | |
| __init__ | 5/21/2019 10:43 A... | PY File | 0 KB |
| CONTRIBUTING.md | 5/21/2019 10:43 A... | MD File | 1 KB |
| eval_util | 5/21/2019 10:43 A... | PY File | 42 KB |
| eval_util_test | 5/21/2019 10:43 A... | PY File | 13 KB |
| export_inference_graph | 5/21/2019 10:43 A... | PY File | 7 KB |
| export_tflite_graph | 5/21/2019 10:43 A... | PY File | 6 KB |
| export_tflite_ssd_graph_lib | 5/21/2019 10:43 A... | PY File | 14 KB |
| export_tflite_ssd_graph_lib_test | 5/21/2019 10:43 A... | PY File | 18 KB |
| exporter | 5/21/2019 10:43 A... | PY File | 20 KB |

Figura 11. API de detección de objetos de TensorFlow

Para utilizar la API de detección de objetos de TensorFlow es necesario instalar Python y configurar un nuevo entorno virtual, también es necesario instalar los siguientes paquetes: protobuf, pillow, lxml, cython, contextlib2, jupyter, matplotlib, pandas, opencv, una vez instalado todos los paquetes se debe crear una variable de entorno con el nombre “PYTHONPATH” que apunte a los directorios de la API de detección de objetos: models, research, slim, asimismo es necesario compilar los archivos protobuf para transformar archivos .proto a archivos .py para que sean compatibles con TensorFlow. Finalmente, mediante línea de comandos ejecutamos el archivo setup.py, este archivo instala la API de detección de objetos de TensorFlow.

3.4.4. Etiquetado de imágenes (Dataset)

Las imágenes recolectadas en la sección 3.4.1 son el conjunto de datos que se serán utilizados para el entrenamiento y validación del modelo.

Las imágenes deben ser etiquetadas con el nombre de la plaga y marcar las coordenadas de la plaga que queremos que detecte, para ello existe un software libre

llamado “LabelImg” que permite realizar el etiquetado de imágenes de manera fácil y efectiva, este software nos genera un archivo XML con la información requerida.

En la **Figura 12,13,14** se muestra el proceso de etiquetado de imágenes del conjunto de datos a través del software LabelImg.

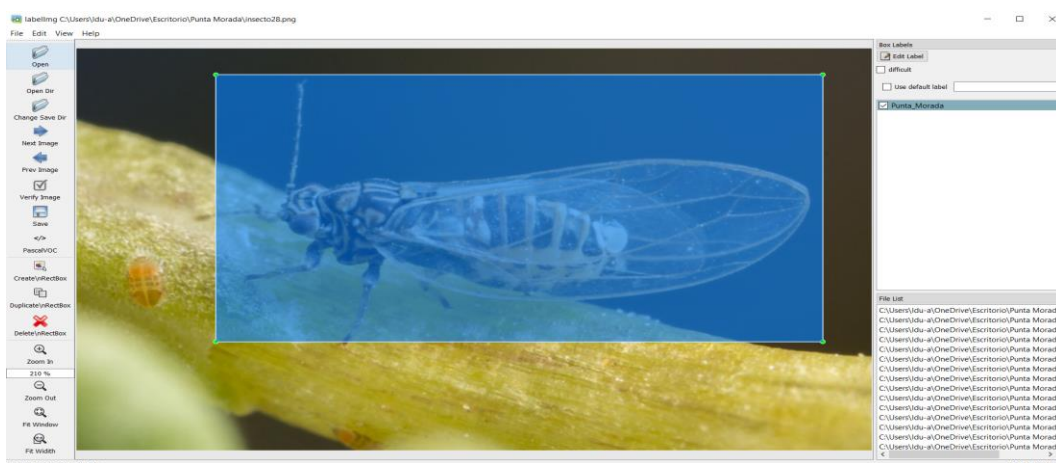


Figura 12. Etiquetado del insecto de la plaga punta morada

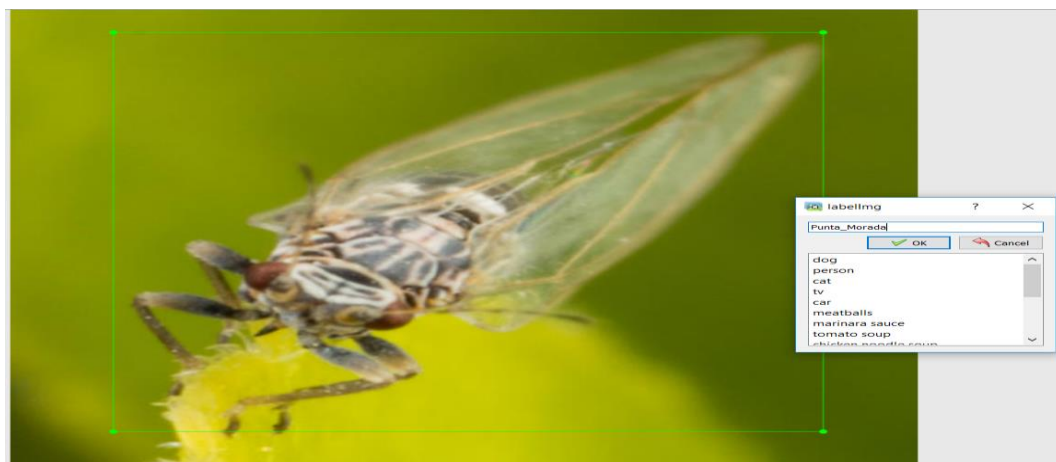


Figura 13. Etiquetado del insecto de la plaga punta morada



Figura 14. Etiquetado del insecto de la plaga mosca minadora

En la **Figura 15** se muestra un archivo con extensión XML con la información obtenida luego de realizar el proceso de etiquetado de una imagen del conjunto de datos.

```

<annotation>
  <folder>Sintomas</folder>
  <filename>sintoma2.jpg</filename>
  <path>C:\Users\Alex Reyes Negrete\Desktop\Investigacion
TEsis\Dataset\Mosca Minadora\Sintomas\sintoma2.jpg</path>
  <source>
    <database>Unknown</database>
  </source>
  <size>
    <width>800</width>
    <height>530</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
  <object>
    <name>minador</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
      <xmin>131</xmin>
      <ymin>118</ymin>
      <xmax>559</xmax>
      <ymax>467</ymax>
    </bndbox>
  </object>
</annotation>

```

Figura 15. Información del resultado de etiquetado de imagen

En la **Figura 16** se muestra la información en formato XML por cada imagen etiquetada del conjunto de datos.

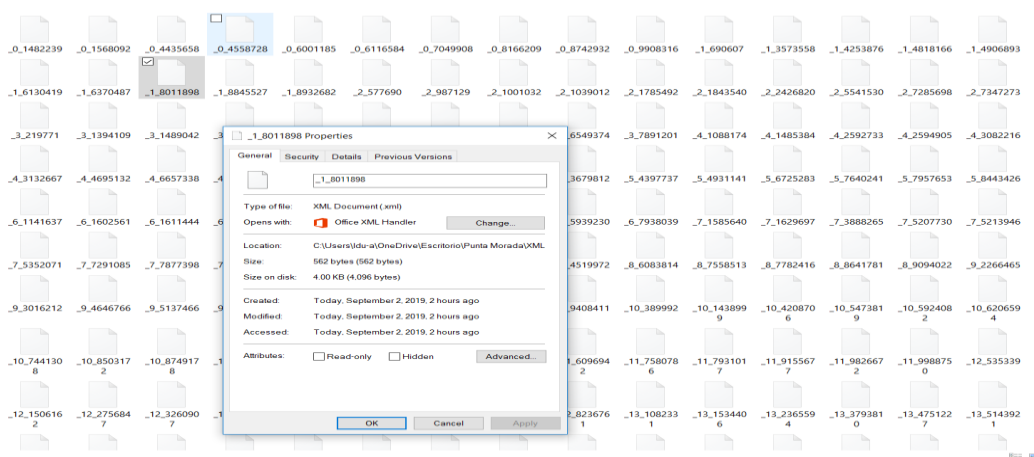


Figura 16. Archivos XML del conjunto de datos

Los archivos XML generados contienen información de la imagen como, nombre de la carpeta donde está almacenada, nombre de la imagen, la ubicación, tamaño y las coordenadas donde se encuentra el objeto etiquetado, en nuestro caso el objeto es el insecto adulto que produce la plaga.

Los archivos XML son convertidos a archivos de texto “CSV” que almacena los datos en forma de columnas, separadas por comas y las filas se distinguen por saltos de línea.

En la **Figura 17** se muestra la información del conjunto de datos en formato CSV

| | |
|-----|--|
| 659 | _125_6087933.png,400,500,minador,16,24,400,456 |
| 660 | _125_67440.png,400,500,minador,7,75,399,432 |
| 661 | _125_7697203.png,400,500,minador,1,58,400,400 |
| 662 | _125_9261570.png,400,500,minador,1,35,345,480 |
| 663 | _126_2033730.png,400,500,minador,20,91,370,469 |
| 664 | _126_2578651.png,400,500,minador,23,11,400,478 |
| 665 | _126_4085764.png,400,500,minador,1,36,371,404 |
| 666 | _126_527645.png,400,500,minador,4,31,375,416 |
| 667 | _126_5750189.png,400,500,minador,5,50,397,432 |
| 668 | _126_6147848.png,400,500,minador,39,61,380,494 |
| 669 | _126_6168977.png,400,500,minador,11,55,377,477 |
| 670 | _126_8200888.png,400,500,minador,32,9,392,457 |
| 671 | _126_9236494.png,400,500,minador,44,86,399,500 |
| 672 | _126_9924550.png,400,500,minador,2,77,400,453 |
| 673 | _127_1465439.png,400,500,minador,87,1,399,500 |
| 674 | _127_2086192.png,400,500,minador,4,43,399,415 |
| 675 | _127_3163599.png,400,500,minador,18,27,373,450 |
| 676 | _127_3811407.png,400,500,minador,24,9,341,500 |
| 677 | _127_4171793.png,400,500,minador,19,2,383,466 |
| 678 | _127_5126676.png,400,500,minador,7,4,355,497 |
| 679 | _127_6069529.png,400,500,minador,9,18,370,436 |
| 680 | _127_7878190.png,400,500,minador,11,2,360,493 |
| 681 | _127_869966.png,400,500,minador,2,21,400,459 |
| 682 | _127_9646882.png,400,500,minador,1,4,400,437 |
| 683 | _128_2482878.png,400,500,minador,1,5,338,500 |
| 684 | _128_2636105.png,400,500,minador,1,27,342,500 |
| 685 | _128_2795693.png,400,500,minador,1,15,329,487 |
| 686 | _128_3437107.png,400,500,minador,1,4,397,500 |
| 687 | _128_428840.png,400,500,minador,1,77,389,500 |

Figura 17. Archivos CSV del conjunto de datos

El archivo CSV es convertido a formato TFRecord. Los archivos en formato TFRecord, son una representación binaria de sus datos o imágenes junto con sus etiquetas, nombres de archivos y otra información adicional. Sirven como datos de entrada para transmitir datos al modelo de red neuronal de manera eficiente mediante el uso de tensores.

En la **Figura 18** se muestran los Archivos TFRecord de entrenamiento y pruebas compatibles con la API de detección de objetos.

| | | | | |
|-------------------------------------|-----------------|-------------------|-----------------------|-----------|
| <input type="checkbox"/> | label_map.pbtxt | 9/3/2019 12:12 PM | PBTEXT File | 1 KB |
| <input checked="" type="checkbox"/> | test.record | 9/9/2019 12:54 PM | RECORD File | 4,035 KB |
| <input checked="" type="checkbox"/> | test_labels | 9/9/2019 12:21 PM | Archivo de valores... | 3 KB |
| <input type="checkbox"/> | train.record | 9/9/2019 12:53 PM | RECORD File | 29,935 KB |
| <input checked="" type="checkbox"/> | train_labels | 9/9/2019 12:21 PM | Archivo de valores... | 20 KB |

Figura 18. Archivos TFRecord

3.4.5. Preparación de los archivos de configuración del modelo de red neuronal

En la **Figura 19** se visualizan los archivos del modelo de red neuronal basado en la arquitectura MobileNet.

ssdlite_mobilenet_v2_coco

| <input type="checkbox"/> | Name | Date modified | Type | Size |
|-------------------------------------|--------------------------------|-------------------|----------------------|-----------|
| <input type="checkbox"/> | saved_model | 9/9/2019 12:56 PM | File folder | |
| <input checked="" type="checkbox"/> | checkpoint | 5/9/2018 11:51 PM | File | 1 KB |
| <input type="checkbox"/> | frozen_inference_graph.pb | 5/9/2018 11:52 PM | PB File | 19,445 KB |
| <input type="checkbox"/> | model.ckpt.data-00000-of-00001 | 5/9/2018 11:51 PM | DATA-00000-OF-0... | 17,779 KB |
| <input type="checkbox"/> | model.ckpt.index | 5/9/2018 11:51 PM | INDEX File | 18 KB |
| <input type="checkbox"/> | model.ckpt.meta | 5/9/2018 11:51 PM | META File | 3,580 KB |
| <input type="checkbox"/> | pipeline | 5/9/2018 11:51 PM | XML Configuration... | 5 KB |

Figura 19. Modelo pre-entrenado con arquitectura MobileNet

Para el proceso de entrenamiento de la red neuronal, la API de detección de objetos está instalado en un servicio de almacenamiento en la nube denominado Google Drive.

También contiene el modelo de red neuronal MobileNet_v2_coco, de igual manera los archivos TFRecord de entrenamiento y validación, que sirven como datos de entrada para la red neuronal.

En la **Figura 20** se visualiza los archivos de entrenamiento y pruebas para el modelo de red neuronal configurados en el entorno de trabajo.

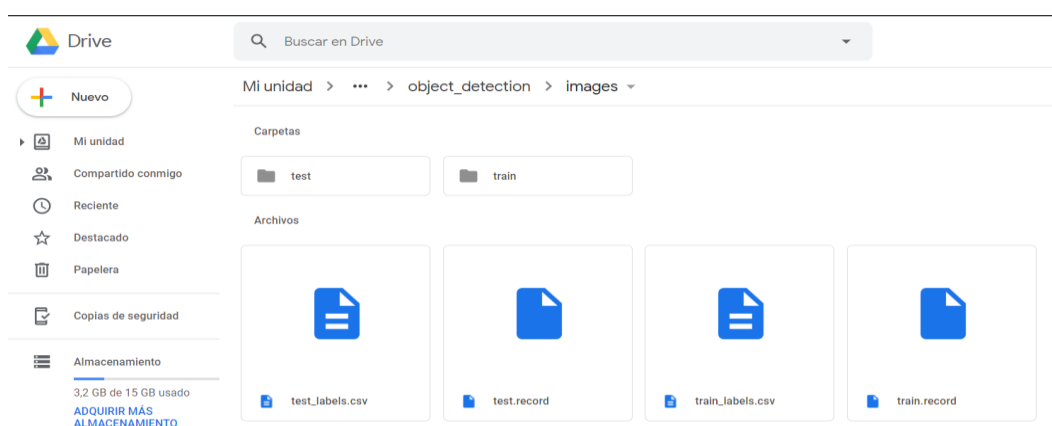


Figura 20. Archivos TFRecord de entrenamiento y pruebas para la red neuronal

El archivo de configuración es muy necesario para que la red neuronal pueda entrenarse, este archivo contiene toda la información acerca de la red neuronal como; el número de clases que la red neuronal al finalizar el proceso de entrenamiento debe detectar, el tamaño de las imágenes de entrenamiento y validación, estructura MobileNet del modelo, el número de iteraciones o épocas en que se va entrenar la red neuronal, el batch size que indica que en cada iteración cuantas imágenes va a tomar del conjunto de entrenamiento y validación, la ubicación donde se encuentra los archivos TFRecord de

entrenamiento, validación, y la ubicación del archivo de texto donde se especifica el número de clases a detectar.

En la **Figura 21** se muestra el archivo de configuración del modelo, donde se debe configurar las rutas a los archivos necesarios para el entrenamiento del modelo de red neuronal, así como el número de pasos y batch size.

```

/
fine_tune_checkpoint: "/content/drive/My Drive/TFConfig/TFmodels/research/object_detection/ssdlite_mobilenet_v2_coco/model.ckpt"
fine_tune_checkpoint_type: "detection"
# Note: The below line limits the training process to 200K steps, which we
# empirically found to be sufficient enough to train the pets dataset. This
# effectively bypasses the learning rate schedule (the learning rate will
# never decay). Remove the below line to train indefinitely.
num_steps: 200000
data_augmentation_options {
  random_horizontal_flip {
  }
}
data_augmentation_options {
  ssd_random_crop {
  }
}
}

train_input_reader: {
  tf_record_input_reader {
    input_path: "/content/drive/My Drive/TFConfig/TFmodels/research/object_detection/images/train.record"
  }
  label_map_path: "/content/drive/My Drive/TFConfig/TFmodels/research/object_detection/training/labelmap.pbtxt"
}

eval_config: {
  num_examples: 850
  # Note: The below line limits the evaluation process to 10 evaluations.
  # Remove the below line to evaluate indefinitely.
  max_evals: 10
}

eval_input_reader: {
  tf_record_input_reader {
    input_path: "/content/drive/My Drive/TFConfig/TFmodels/research/object_detection/images/test.record"
  }
  label_map_path: "/content/drive/My Drive/TFConfig/TFmodels/research/object_detection/training/labelmap.pbtxt"
  shuffle: false
  num_readers: 1
}

```

Figura 21. Archivo de configuración del modelo

En la **Figura 22** se muestra el archivo de texto con las etiquetas definidas para el modelo de red neuronal.

```

item {
  id: 1
  name: 'minador'
}

item {
  id: 2
  name: 'IPuntaMorada'
}

item {
  id: 3
  name: 'insecto_normal'
}



```

Figura 22. Etiquetas del modelo de red neuronal

A continuación, en la tabla 10 se detalla las etiquetas definidas para el modelo de red neuronal.

Tabla 3

Nombres de etiquetas para el entrenamiento del modelo

| Plaga | Etiqueta | Imagen |
|-----------------------|--------------|---|
| Punta Morada | IPuntaMorada |  |
| Mosca Minadora | minador |  |

3.4.6. Entrenamiento y pruebas del modelo de red neuronal

Para la ejecución del modelo se creó un cuaderno de ejecución en Google Colaboratory, y de igual manera se vincula al Google Drive donde se encuentran todos los archivos necesarios para el entrenamiento y validación de la red neuronal. En la **Figura 23** se muestra el código para vincular el servicio de almacenamiento de Google drive hacia el entorno de trabajo Google Colaboratory.

```
▶ from os.path import join
   from google.colab import drive

   ROOT = "/content/drive"
   drive.mount(ROOT)

   PROJ = "My Drive/TFConfig" # This is a custom path.
   PROJECT_PATH = join(ROOT, PROJ)

   %cd ~/content
   %cd drive/My Drive/TFConfig
```

Figura 23. Código para vincular el archivo Colaboratory a Google Drive

En los archivos de la API de detección de objetos de TensorFlow en la carpeta Legacy se encuentra el archivo de Python para ejecutar el entrenamiento del modelo, en el archivo de Google Colaboratory accedemos a la ubicación de dicha carpeta y ejecutamos el archivo train.py, especificando el directorio donde se va a guardar cada 5 minutos los tres archivos que guardan los valores de los puntos de control. También se especifica la carpeta donde se encuentra el archivo de configuración del modelo.

En la **Figura 24** se muestra el comando para iniciar el proceso de entrenamiento, teniendo como parámetros de entrada la carpeta con los archivos de configuración del modelo de red neuronal.

```
!python train.py --logtostderr --train_dir=training/ --pipeline_config_path=training/ssd_mobilenet_v2_coco.config
```

Figura 24. Código para ejecutar el entrenamiento del modelo

El proceso de entrenamiento inicia de acuerdo al archivo de configuración creado para el modelo, este archivo especifica que el número de iteraciones que se va a realizar es de 200000. También se puede observar la función de pérdida comienza muy alto aproximadamente en un valor de 20, a medida que avanza el entrenamiento la función de pérdida va disminuyendo.

También el proceso de entrenamiento va conjuntamente con el de validación debido a que la API de detección de objetos no divide el proceso de entrenamiento y validación por separado.

En la **Figura 25** se puede observar el inicio del entrenamiento del modelo desde el paso 1 y con una pérdida de alrededor de 20.

```

I0924 16:16:49.478669 140232521619200 supervisor.py:1099] global_step/sec: 0
2019-09-24 16:16:53.000997: I tensorflow/stream_executor/platform/default/dso_loader.cc:42] Successfully opened dynamic library libcudnn.so.7
INFO:tensorflow:Recording summary at step 0.
I0924 16:16:58.774388 140232513226496 supervisor.py:1050] Recording summary at step 0.
INFO:tensorflow:global step 1: loss = 27.9845 (30.945 sec/step)
I0924 16:17:00.245972 140235584698240 learning.py:507] global step 1: loss = 27.9845 (30.945 sec/step)
INFO:tensorflow:global step 2: loss = 23.1356 (3.970 sec/step)
I0924 16:17:05.089189 140235584698240 learning.py:507] global step 2: loss = 23.1356 (3.970 sec/step)
INFO:tensorflow:global step 3: loss = 21.7948 (2.232 sec/step)
I0924 16:17:07.485946 140235584698240 learning.py:507] global step 3: loss = 21.7948 (2.232 sec/step)
INFO:tensorflow:global step 4: loss = 21.3653 (1.616 sec/step)
I0924 16:17:09.104442 140235584698240 learning.py:507] global step 4: loss = 21.3653 (1.616 sec/step)
2019-09-24 16:17:09.156024: W tensorflow/core/framework/allocator.cc:107] Allocation of 1617225120 exceeds 10% of system memory.
tcmalloc: large alloc 1617231872 bytes == 0x1500d8000 @ 0x7f8b23f0db6b 0x7f8b23f2d379 0x7f8ae630407 0x7f8ae630407 0x7f8ae630407 0x7f8ae630407
INFO:tensorflow:global step 5: loss = 20.2767 (1.919 sec/step)
I0924 16:17:11.025228 140235584698240 learning.py:507] global step 5: loss = 20.2767 (1.919 sec/step)
INFO:tensorflow:global step 6: loss = 19.6046 (1.519 sec/step)
I0924 16:17:12.546729 140235584698240 learning.py:507] global step 6: loss = 19.6046 (1.519 sec/step)
INFO:tensorflow:global step 7: loss = 19.2709 (1.923 sec/step)
I0924 16:17:14.471805 140235584698240 learning.py:507] global step 7: loss = 19.2709 (1.923 sec/step)
INFO:tensorflow:global step 8: loss = 18.8010 (1.733 sec/step)
I0924 16:17:16.206661 140235584698240 learning.py:507] global step 8: loss = 18.8010 (1.733 sec/step)
INFO:tensorflow:global step 9: loss = 17.8866 (1.646 sec/step)
I0924 16:17:17.854629 140235584698240 learning.py:507] global step 9: loss = 17.8866 (1.646 sec/step)
INFO:tensorflow:global step 10: loss = 17.8705 (1.824 sec/step)
I0924 16:17:19.680182 140235584698240 learning.py:507] global step 10: loss = 17.8705 (1.824 sec/step)
INFO:tensorflow:global step 11: loss = 17.6323 (2.011 sec/step)
I0924 16:17:21.692838 140235584698240 learning.py:507] global step 11: loss = 17.6323 (2.011 sec/step)

```

Figura 25. Inicio de entrenamiento del modelo

En la **Figura 26** se muestra la disminución de la función de pérdida a medida que aumenta el número de pasos de entrenamiento.

```

INFO:tensorflow:global step 18: loss = 9.5413 (2.359 sec/step)
I0919 17:44:37.968914 139964043749248 learning.py:507] global step 18: loss = 9.5413 (2.359 sec/step)
INFO:tensorflow:global step 19: loss = 8.9571 (2.105 sec/step)
I0919 17:44:40.076429 139964043749248 learning.py:507] global step 19: loss = 8.9571 (2.105 sec/step)
INFO:tensorflow:global step 20: loss = 8.8562 (1.506 sec/step)
I0919 17:44:41.583987 139964043749248 learning.py:507] global step 20: loss = 8.8562 (1.506 sec/step)
INFO:tensorflow:global step 21: loss = 9.4734 (1.830 sec/step)
I0919 17:44:43.416616 139964043749248 learning.py:507] global step 21: loss = 9.4734 (1.830 sec/step)
INFO:tensorflow:global step 22: loss = 8.1302 (1.579 sec/step)
I0919 17:44:44.997965 139964043749248 learning.py:507] global step 22: loss = 8.1302 (1.579 sec/step)
INFO:tensorflow:global step 23: loss = 8.6663 (1.702 sec/step)
I0919 17:44:46.701590 139964043749248 learning.py:507] global step 23: loss = 8.6663 (1.702 sec/step)
INFO:tensorflow:global step 24: loss = 8.1676 (1.949 sec/step)
I0919 17:44:48.653000 139964043749248 learning.py:507] global step 24: loss = 8.1676 (1.949 sec/step)
INFO:tensorflow:global step 25: loss = 8.6572 (1.900 sec/step)
I0919 17:44:50.555161 139964043749248 learning.py:507] global step 25: loss = 8.6572 (1.900 sec/step)
INFO:tensorflow:global step 26: loss = 7.8692 (1.650 sec/step)
I0919 17:44:52.208496 139964043749248 learning.py:507] global step 26: loss = 7.8692 (1.650 sec/step)
INFO:tensorflow:global step 27: loss = 7.9861 (1.626 sec/step)
I0919 17:44:53.836239 139964043749248 learning.py:507] global step 27: loss = 7.9861 (1.626 sec/step)
INFO:tensorflow:global step 28: loss = 7.7359 (1.679 sec/step)
I0919 17:44:55.517257 139964043749248 learning.py:507] global step 28: loss = 7.7359 (1.679 sec/step)
INFO:tensorflow:global step 29: loss = 8.0817 (1.653 sec/step)
I0919 17:44:57.172676 139964043749248 learning.py:507] global step 29: loss = 8.0817 (1.653 sec/step)
INFO:tensorflow:global step 30: loss = 7.5046 (1.509 sec/step)
I0919 17:44:58.683725 139964043749248 learning.py:507] global step 30: loss = 7.5046 (1.509 sec/step)
INFO:tensorflow:global step 31: loss = 7.2597 (1.482 sec/step)
I0919 17:45:00.168510 139964043749248 learning.py:507] global step 31: loss = 7.2597 (1.482 sec/step)
INFO:tensorflow:global step 32: loss = 7.5410 (1.644 sec/step)
I0919 17:45:01.814765 139964043749248 learning.py:507] global step 32: loss = 7.5410 (1.644 sec/step)

```

Figura 26. Disminución de la función de pérdida

El proceso de entrenamiento del modelo de red neuronal con arquitectura MobileNet duro aproximadamente 50 horas, en el transcurso del entrenamiento la función de pérdida disminuye consistentemente por debajo de 1.

En la **Figura 27** se visualiza el proceso de entrenamiento, teniendo una perdida por debajo de 2 en 90000 iteraciones realizadas.

```

INFO:tensorflow:global step 92282: loss = 0.9886 (1.787 sec/step)
I0924 00:00:18.554979 139713348188032 learning.py:507] global step 92282: loss = 0.9886 (1.787 sec/step)
...
INFO:tensorflow:global step 92283: loss = 1.5975 (1.806 sec/step)
I0924 00:00:20.362720 139713348188032 learning.py:507] global step 92283: loss = 1.5975 (1.806 sec/step)
INFO:tensorflow:global step 92284: loss = 1.4835 (1.561 sec/step)
I0924 00:00:21.925165 139713348188032 learning.py:507] global step 92284: loss = 1.4835 (1.561 sec/step)
INFO:tensorflow:global step 92285: loss = 1.4320 (1.758 sec/step)
I0924 00:00:23.684650 139713348188032 learning.py:507] global step 92285: loss = 1.4320 (1.758 sec/step)
INFO:tensorflow:global step 92286: loss = 1.9679 (1.499 sec/step)
I0924 00:00:25.185527 139713348188032 learning.py:507] global step 92286: loss = 1.9679 (1.499 sec/step)
INFO:tensorflow:global step 92287: loss = 0.9885 (1.363 sec/step)
I0924 00:00:26.550363 139713348188032 learning.py:507] global step 92287: loss = 0.9885 (1.363 sec/step)
INFO:tensorflow:global step 92288: loss = 1.6901 (1.502 sec/step)
I0924 00:00:28.054518 139713348188032 learning.py:507] global step 92288: loss = 1.6901 (1.502 sec/step)
INFO:tensorflow:global step 92289: loss = 1.3794 (1.420 sec/step)
I0924 00:00:29.476471 139713348188032 learning.py:507] global step 92289: loss = 1.3794 (1.420 sec/step)
INFO:tensorflow:global step 92290: loss = 1.6796 (1.462 sec/step)
I0924 00:00:30.941288 139713348188032 learning.py:507] global step 92290: loss = 1.6796 (1.462 sec/step)
INFO:tensorflow:global step 92291: loss = 1.2946 (1.679 sec/step)
I0924 00:00:32.622629 139713348188032 learning.py:507] global step 92291: loss = 1.2946 (1.679 sec/step)
INFO:tensorflow:global step 92292: loss = 1.3116 (1.804 sec/step)
I0924 00:00:34.428700 139713348188032 learning.py:507] global step 92292: loss = 1.3116 (1.804 sec/step)
INFO:tensorflow:global step 92293: loss = 1.4028 (1.592 sec/step)
I0924 00:00:36.024017 139713348188032 learning.py:507] global step 92293: loss = 1.4028 (1.592 sec/step)

```

Figura 27. Proceso de entrenamiento

La función de pérdida del modelo con el conjunto de dataset entrenado alcanzo un valor aproximado de 0.70 en 200000 iteraciones, las cuales se detallan en la siguiente tabla.

Tabla 4

Función de perdida por cada 10000 iteraciones

| No | Iteraciones | Valor aproximado de la función de perdida |
|----|-------------|---|
| 1 | 10000 | 4.1492 |
| 2 | 20000 | 3.6284 |
| 3 | 30000 | 2.9873 |
| 4 | 40000 | 2.6856 |
| 5 | 50000 | 2.4263 |
| 6 | 60000 | 2.0256 |
| 7 | 70000 | 1.6796 |
| 8 | 80000 | 1.3794 |
| 9 | 90000 | 0.9885 |

CONTINÚA 

| | | |
|-----------|--------|--------|
| 10 | 100000 | 0.9790 |
| 11 | 110000 | 0.9634 |
| 12 | 120000 | 0.9263 |
| 13 | 130000 | 0.9023 |
| 14 | 140000 | 0.8678 |
| 15 | 150000 | 0.8312 |
| 16 | 160000 | 0.8098 |
| 17 | 170000 | 0.7899 |
| 18 | 180000 | 0.7623 |
| 19 | 190000 | 0.7412 |
| 20 | 200000 | 0.7145 |

Tensorboard es una herramienta de visualización de TensorFlow que permite realizar el seguimiento y visualización de métricas de pérdida y precisión del modelo, también permite observar el gráfico del modelo, ver histogramas de pesos, sesgos u otros tensores que a medida cambian con el tiempo (TensorFlow, 2019).

En la **Figura 28** se puede observar la función de pérdida del modelo al finalizar el entrenamiento; la gráfica es generada por tensorboard¹⁹.

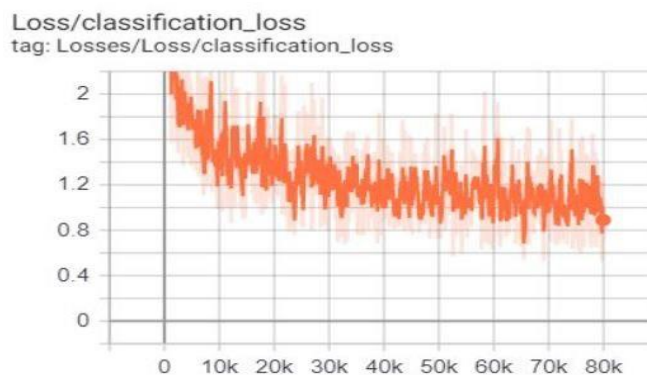


Figura 28. Función de pérdida del modelo

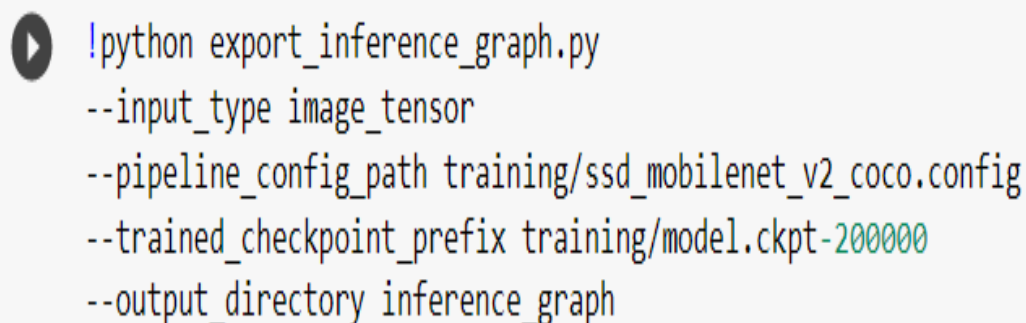
¹⁹ Conjunto de herramientas de visualización de Tensorflow

3.4.7. Validación del modelo de red neuronal

Una vez concluido el proceso de entrenamiento el último paso es generar el archivo congelado (archivo. pb), para esto en el directorio de “object_detection” tenemos el archivo “export_inference_graph” que ayuda a generar el archivo congelado con los pesos del nuevo entrenamiento.

Como se observa en la **Figura 29** se ingresa el archivo de valores correspondientes al último punto de control guardado (archivo. ckpt), en este caso particular “model.ckpt-92317” así como el nombre de la carpeta que se desea guardar el archivo congelado frozen_inference_graph. pb el cual permite realizar diferentes pruebas al modelo para corroborar que la precisión sea la correcta.

En la **Figura 29** se muestra el comando para generar el archivo de inferencia, donde se especifica el archivo de configuración y el número total de pasos con que se realizó en el entrenamiento.

A terminal window showing a command to export an inference graph. The command is: !python export_inference_graph.py --input_type image_tensor --pipeline_config_path training/ssd_mobilenet_v2_coco.config --trained_checkpoint_prefix training/model.ckpt-200000 --output_directory inference_graph. The command is preceded by a play button icon.

```
!python export_inference_graph.py
--input_type image_tensor
--pipeline_config_path training/ssd_mobilenet_v2_coco.config
--trained_checkpoint_prefix training/model.ckpt-200000
--output_directory inference_graph
```

Figura 29. Comando para generar el archivo congelado

En la **Figura 30** se muestran los archivos generados luego de la ejecución del código de la figura 29, el archivo congelado `frozen_inference_graph` generado permite hacer pruebas al modelo desde el computador.

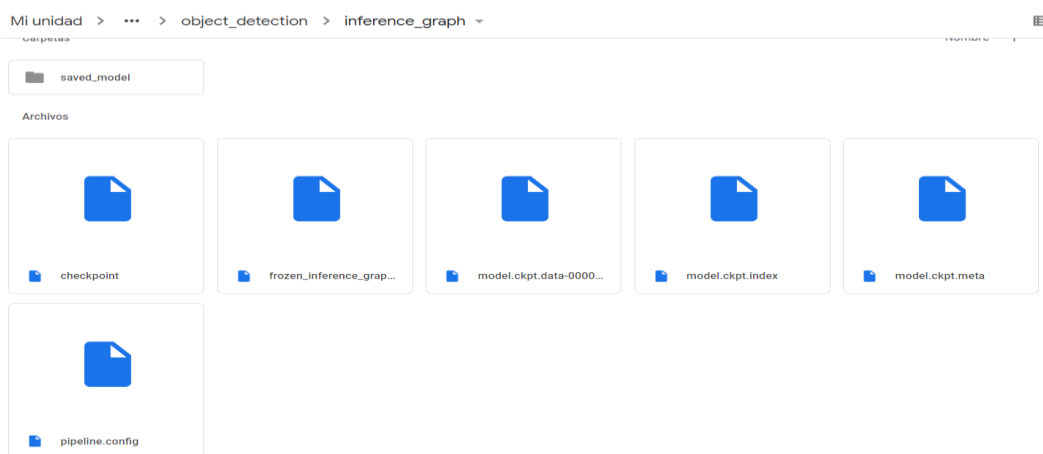


Figura 30. Archivo congelado `frozen_inference_graph`

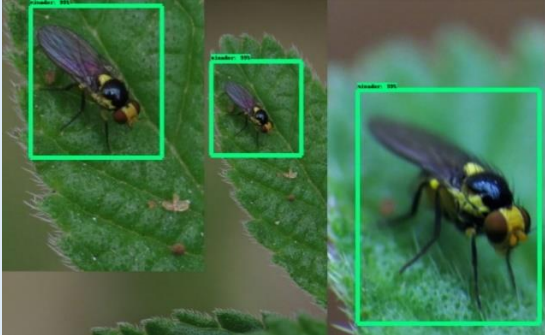



El archivo congelado `frozen_inference_graph` permite realizar pruebas al modelo de red neuronal y comprobar que los resultados de predicción son los esperados, de igual manera es necesario el archivo `labelmap.pbtxt` que contiene los nombres de cada clase.

Para las pruebas de detección se probó con las dos plagas y diferentes niveles de iluminación y tamaño, así como también con imágenes diferentes a las que se encontraban dentro de las imágenes de entrenamiento y test.

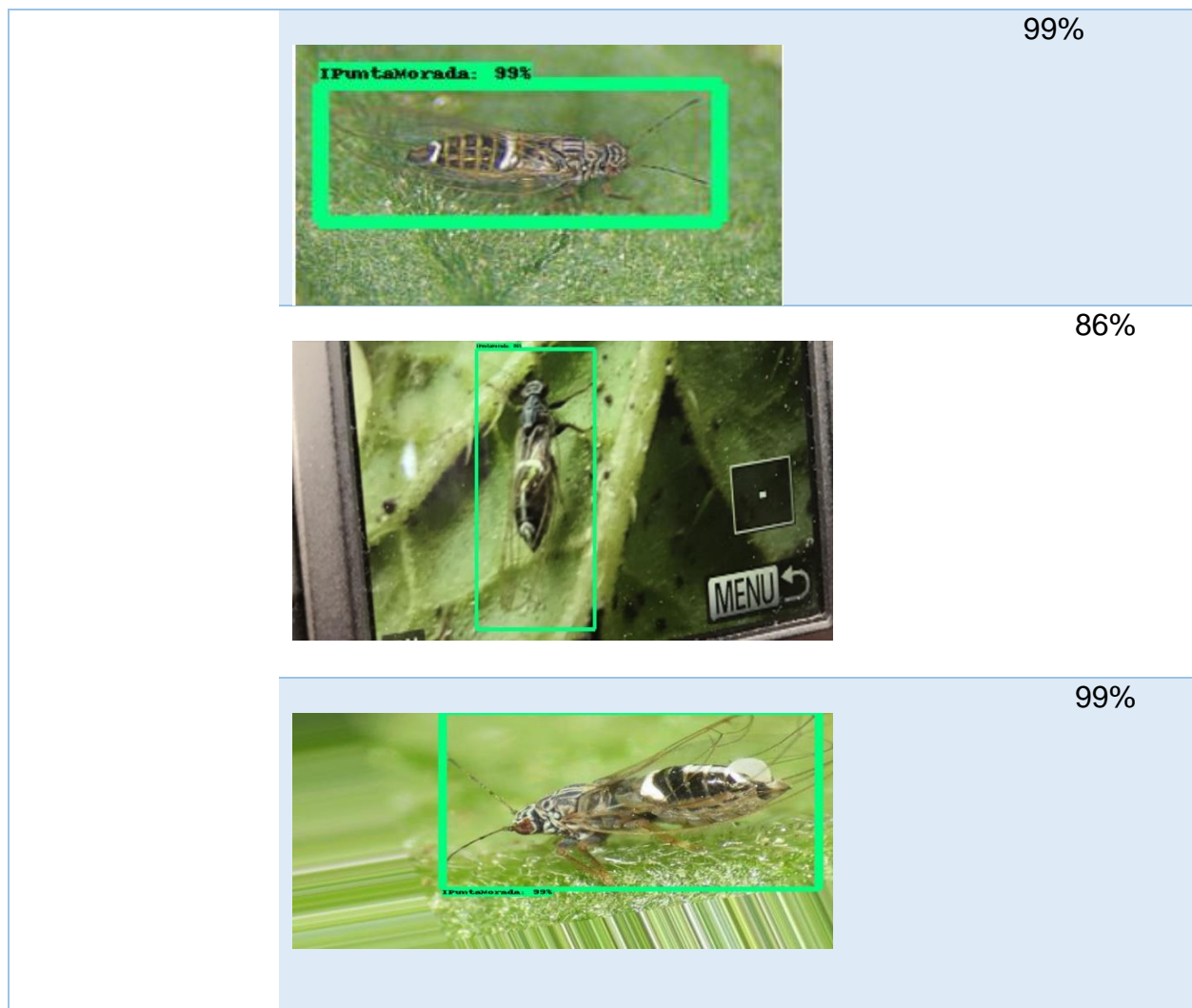
A continuación, se detalla una tabla con las pruebas realizadas al modelo de red neuronal

Tabla 5

Pruebas realizadas al modelo de red neuronal

| N. | Plaga | Diagnóstico | Acierto |
|----|--------------|--|---------|
| 1 | Minador |  | 99% |
| | |  | 99% |
| | |  | 77% |
| 2 | Punta morada |  | 92% |

CONTINÚA 



3.4.8. Convertir el modelo de red neuronal a un formato compatible para dispositivos móviles

Una vez realizado las pruebas al modelo de red neuronal y obteniendo los resultados esperados, se procede a transformar el modelo a una versión para que pueda ser ejecutado desde Android Studio y funcionar a través de la cámara del dispositivo móvil en tiempo real, para esto se hizo uso de TensorFlow Lite que permite obtener un modelo

más pequeño y permite aprovechar las operaciones que se han optimizado para dispositivos móviles.

TensorFlow Lite es la solución ligera de TensorFlow para dispositivos móviles e integrados. Permite la inferencia de aprendizaje automático en el dispositivo con baja latencia y un pequeño tamaño binario. TensorFlow Lite utiliza muchas técnicas para esto, como núcleos cuantizados que permiten modelos más pequeños y más rápidos (Robinson, Chowdhery, & Huang, 2018).

Para convertir el modelo es necesario el archivo congelado (archivo. pb) que permite convertir en un modelo más ligero con extensión. Tflite que contiene todos los archivos necesarios, pero en una versión más liviana.

En la **Figura 31** se muestra el archivo transformado detect. tflite, el cual es compatible con dispositivos móviles.

| Nombre | Fecha de modifica... | Tipo | Tamaño |
|-----------------|----------------------|----------------|-----------|
| detect.tflite | 27/09/2019 2:08 | Archivo TFLITE | 5.514 KB |
| tflite_graph.pb | 25/09/2019 20:37 | Archivo PB | 22.033 KB |
| tflite_graph | 27/09/2019 9:58 | Archivo PBTXT | 24.576 KB |

Figura 31. Archivo detect. tflite

3.5. Desarrollo de la aplicación móvil

3.5.1. Exploración

A continuación se detalla el establecimiento de los interesados y el establecimiento del proyecto de la aplicación móvil, con el fin de obtener todas las bases necesarias para su implementación. Esto asegura el éxito en las próximas fases de la metodología.

a. Establecimiento de los Stakeholders

- Técnico agrónomo a cargo del sector Toacaso: es la persona encargada de realizar el recorrido en los diferentes cultivos del sector con el propósito de realizar el proceso de diagnóstico de plagas en cultivos de papas, con el objetivo de alertar posibles amenazas de plagas a los diferentes agricultores del sector.
- Agricultores del sector: corresponde a las personas propias de la parroquia toacaso sector la libertad que se dedican al cultivo de papas y por consiguiente necesitan conocer sobre las diferentes plagas que pueden afectar sus cultivos

b. Establecimiento del Proyecto

El presente proyecto tiene como objetivo desarrollar una aplicación móvil para dispositivos con sistema operativo Android para optimizar el diagnóstico de plagas en hojas de cultivos de papas en la parroquia Toacaso sector la libertad, facilitando el acceso a la información sobre las diferentes plagas que existen y los diferentes insecticidas que se pueden aplicar para el control de las plagas, así como también el proceso de diagnóstico de las plagas que se va a realizar en tiempo real mediante la cámara del dispositivo móvil de una manera fácil e intuitiva.

A nivel tecnológico el desarrollo del proyecto de investigación es necesarios los recursos mencionados a continuación.

Desde el punto de vista hardware es necesario dispositivos móviles con cámaras de capacidad mínima 13 megapíxeles, aunque se recomienda cámaras de alta calidad, con relación al software los dispositivos deben contar con sistema operativo Android con versión mayor o igual a 7.

Con el propósito de garantizar el correcto funcionamiento de la aplicación y de causar un gran impacto en los agricultores y técnico agrónomo, la aplicación se desarrollará teniendo en cuenta todos los aspectos de usabilidad necesarios para tener una interfaz amigable.

3.5.2. Inicialización

En esta fase se va a detallar los requerimientos iniciales de la aplicación, definición inicial del proyecto, propósito y funcionalidad:

a. Requisitos Iniciales

A continuación, se detallan los requerimientos iniciales de la aplicación:

- Detectar posibles amenazas de plagas en hojas de cultivos de papas.
- Información de las plagas existentes en cultivos de papas
- Información de insecticidas existentes para el control de plagas.
- Información sobre el proceso de diagnóstico de plagas en cultivos de papas.

b. Análisis de requerimientos iniciales

A continuación, se establece el nivel de importancia que tiene cada requerimiento inicial, con un rango establecido de 1 a 10 en el que 10 representa el grado de mayor importancia.

Tabla 6*Requisitos Iniciales*

| Requisitos Iniciales | Importancia |
|--|-------------|
| Detectar posibles amenazas de plagas en hojas de cultivos de papas | 10 |
| Información de las plagas existentes en cultivos de papas | 10 |
| Información de insecticidas existentes para el control de plagas | 10 |
| Información sobre el proceso de Diagnóstico de plagas en cultivos de papas | 8 |

c. Pila del Producto

Con el fin de seguir un proceso ordenado y consecuente de las tareas se han establecido las historias de usuario en la siguiente tabla organizadas de acuerdo con su importancia.

Tabla 7*Pila del Producto*

| Historia N. | Descripción | Prioridad | Estimado en horas | Iteraciones |
|-------------|--|-----------|-------------------|-------------|
| 1 | Detectar posibles amenazas de plagas en hojas de cultivos de papas | Alta | 100 | 2 |
| 2 | Información de las plagas existentes en cultivos de papas | Alta | 25 | 2 |
| 3 | Información de insecticidas existentes para el control de plagas | Alta | 25 | 2 |
| 4 | Información sobre el proceso de Diagnóstico de plagas en cultivos de papas | Alta | 25 | 2 |

d. Módulos de la aplicación móvil**a. Módulo de detección de plagas**

- Detectar posibles amenazas de plagas en hojas de cultivos de papas en tiempo real.

b. Módulo de Información

- Información de las plagas existentes en cultivos de papas
- Información de insecticidas existentes para el control de plagas
- Información sobre el proceso de diagnóstico de plagas en cultivos de papas

e. Limitaciones

- a.** La aplicación móvil solo se puede ejecutar en dispositivos Android versión 6.0 en adelante

f. Supuestos y Dependencias

La información que se encuentran en la aplicación es corroborada por el técnico agrónomo encargado del sector.

Para el correcto funcionamiento de la aplicación los usuarios deben tener conocimientos básicos sobre el uso de smartphone con sistema operativo Android.

Las interfaces de la aplicación se describen en lenguaje español.

g. Planificación Inicial

A continuación, se detallan todas las historias de usuario para el desarrollo de la aplicación

Tabla 8

Historia de Usuario: Detectar posibles amenazas de plagas en hojas de cultivos de papas

| Historia de Usuario | |
|---|---|
| Numero: 1 | Usuario: Técnico agrónomo y agricultores |
| Nombre: Detectar posibles amenazas de plagas en hojas de cultivos de papas | |
| Prioridad: Alta | Riesgo: Alto |
| Puntos: 13 | |
| Programador: Choloquina Luis, Reyes Alex | |
| Descripción: permitir a la aplicación detectar posibles amenazas de plagas en tiempo real mediante el uso de la cámara del dispositivo | |
| Observaciones: Ninguna | |

Tabla 9

Historia de Usuario: Información de las plagas existentes en cultivos de papas

| Historia de Usuario | |
|--|---|
| Numero: 2 | Usuario: Técnico agrónomo y agricultores |
| Nombre: Información de las plagas existentes en cultivos de papas | |
| Prioridad: Alta | Riesgo: Medio |
| Puntos: 8 | |
| Programador: Choloquina Luis, Reyes Alex | |
| Descripción: permitir a la aplicación mostrar información sobre las plagas que amenazan a los cultivos de papas | |

CONTINÚA 

Observaciones: **Ninguna**

Tabla 10

Historia de Usuario: Información de insecticidas existentes para el control de plagas.

| Historia de Usuario | |
|--|---|
| Numero: 3 | Usuario: Técnico agrónomo y agricultores |
| Nombre: Información de insecticidas existentes para el control de plagas | |
| Prioridad: Alta | Riesgo: Medio |
| Puntos: 8 | |
| Programador: Choloquina Luis, Reyes Alex | |
| Descripción: permitir a la aplicación mostrar información sobre los insecticidas adecuados para el control preventivo de plagas | |
| Observaciones: Ninguna | |

Tabla 11

Historia de Usuario: Información sobre el proceso de Diagnóstico de plagas en cultivos de papas.

| Historia de Usuario | |
|--|---|
| Numero: 4 | Usuario: Técnico agrónomo y agricultores |
| Nombre: Información sobre el proceso de Diagnóstico de plagas en cultivos de papas. | |
| Prioridad: Alta | Riesgo: Medio |
| Puntos: 8 | |
| Programador: Choloquina Luis, Reyes Alex | |

CONTINÚA 

Descripción: **permitir a la aplicación mostrar información sobre el correcto proceso de diagnóstico de plagas en cultivos de papas.**

Observaciones: **Ninguna**

h. Análisis de requerimientos y pre-requisitos

Con el objetivo de cumplir con todos los requerimientos funcionales de la aplicación es necesario definir los pre-requisitos a nivel técnico que permitan realizar la implementación de cada requerimiento.

Detectar posibles amenazas de plagas en hojas de cultivos de papas

Pre-requisitos

- Modelo de red neuronal entrenado y validado compatible con dispositivos android unido a la aplicación.
- Activar la cámara del dispositivo móvil.

Información de las plagas existentes en cultivos de papas

Pre-requisitos

- Información validada por el agrónomo

Información de insecticidas existentes para el control de plagas

Pre-requisitos

- Información validada por el agrónomo

Información sobre el proceso de Diagnóstico de plagas en cultivos de papas

Pre-requisitos

- Información validada por el agrónomo

i. Planificación de fases

Tabla 12

Planificación de fases

| Fase | Iteración | Descripción |
|-----------------------|---|--|
| Inicialización | Iteración 0 | Establecimiento del proyecto, Análisis de requerimientos iniciales, Supuestos y dependencias |
| Producción | Iteración módulo de detección de plagas | Implementación del módulo de detección de plagas, mejora y actualización de historias de usuarios, Refinamiento de interfaces, Creación y cumplimiento de pruebas de aceptación. |
| | Iteración módulo de información | Implementación del módulo de información mejora y actualización de historias de usuarios, Refinamiento de interfaces, Creación y cumplimiento de pruebas de aceptación. |
| Estabilización | Iteración módulo de detección de plagas | Mejora del módulo de detección de plagas, Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación. |
| | Iteración módulo de información | Mejora del módulo de detección de información, Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación. |

CONTINÚA 

| | | |
|----------------------------|----------------------------------|--|
| Pruebas del Sistema | Iteración de pruebas del sistema | Se realiza la evaluación de las pruebas y se ejecuta el análisis de los resultados |
|----------------------------|----------------------------------|--|

j. Diseño de la aplicación

En la **Figura 32** se muestra el diseño de la aplicación, el cual está conformado por dos componentes principales, un modelo de red neuronal entrenado y validado, en un formato compatible para dispositivos móviles y una aplicación móvil.

Esta aplicación móvil utiliza el modelo de red neuronal para realizar el proceso de diagnóstico de plagas en tiempo real a través del acceso a la cámara del dispositivo móvil.

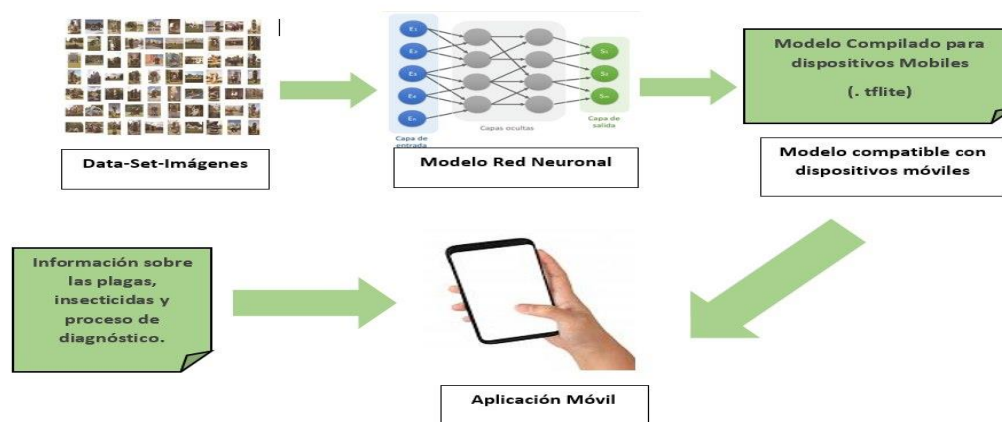


Figura 32. Diseño de la aplicación

k. Arquitectura de la aplicación

En la **Figura 33** se presenta la arquitectura de la aplicación, conformado por la aplicación móvil y un modelo de red neuronal convolucional, el cual es unido dentro de la aplicación móvil para que funcione a través de la cámara del dispositivo móvil.

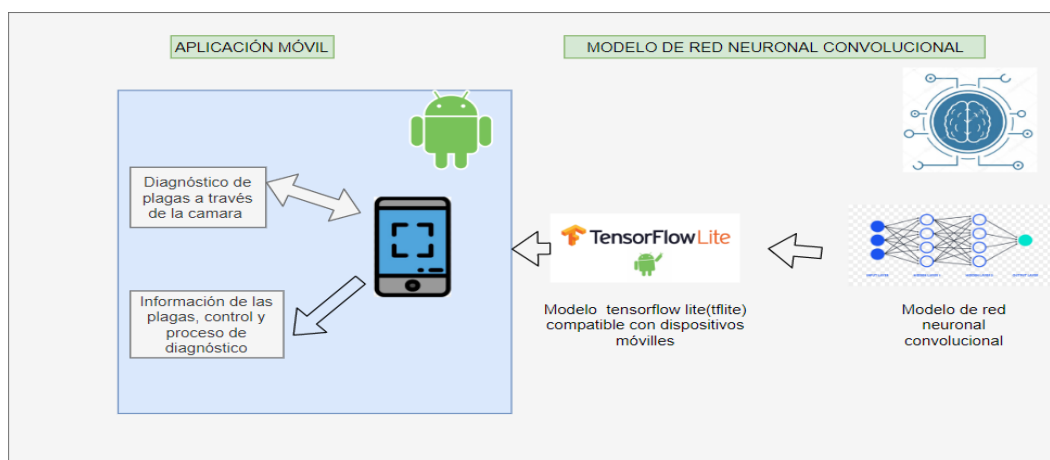


Figura 33. Arquitectura de la aplicación

I. Descripción de las Interfaces de Usuario

La aplicación móvil cuenta con varias interfaces amigables para el usuario, que permiten acceder a todas las funcionalidades de la aplicación.

Tabla 13

Icono de la aplicación



| Interfaz | Descripción |
|---|---|
|  | <ul style="list-style-type: none"> Una vez instalada la aplicación en el dispositivo móvil se podrá visualizar con el nombre de PlagasApp y con su logo correspondiente. |

Tabla 14
Pantallas de bienvenida

| Interfaz | Descripción |
|---|--|
|  | <ul style="list-style-type: none">• Esta es la primera pantalla que se muestra al ejecutar la aplicación |
|  | <ul style="list-style-type: none">• Esta es la pantalla de bienvenida a la aplicación, cuenta con un botón en la parte inferior derecha que permite saltar directamente a la pantalla principal. |

- Esta es la pantalla informativa de la aplicación



- Esta es una pantalla informativa de la aplicación




Tabla 15
Menú Principal

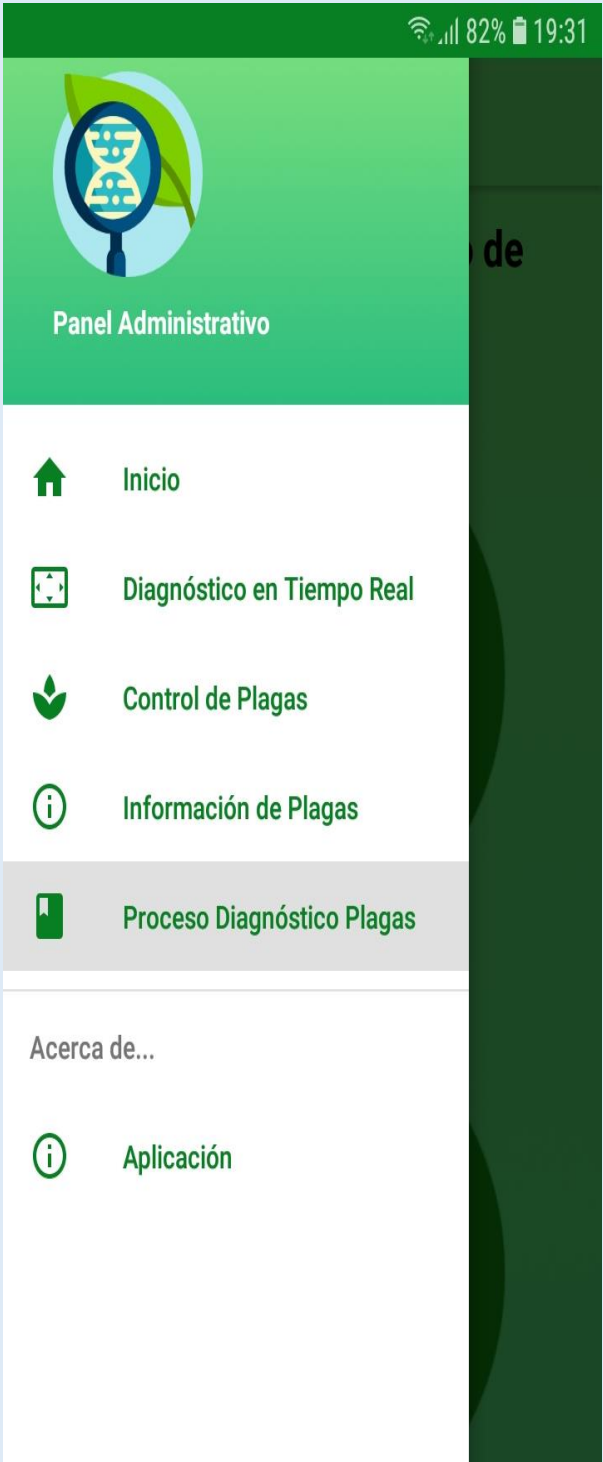
| Interfaz | Descripción |
|--|--|
|  | <ul style="list-style-type: none"> • Esta pantalla muestra el menú principal de la aplicación con el acceso a todas las funcionalidades de la aplicación. • Inicio: permite acceder a la vista principal. • Diagnóstico en tiempo real: permite realizar un diagnóstico de plagas en tiempo real a través de la cámara del dispositivo móvil. • Control de plagas: permite visualizar la información sobre los insecticidas para el control de plagas. • Información de plagas: permite visualizar información sobre las diferentes plagas como punta morada y mosca minadora de la papa. • Proceso Diagnóstico Plagas: muestra el proceso correcto para realizar el diagnóstico de plagas en su cultivo. |

Tabla 16
Pantalla de Inicio

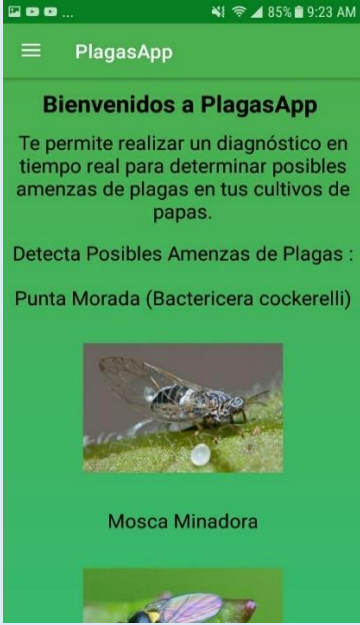
| Interfaz | Descripción |
|--|---|
|  | <ul style="list-style-type: none"> • Esta pantalla permite visualizar información sobre las funcionalidades que permite realizar la aplicación |

Tabla 17
Pantalla Diagnóstico en Tiempo Real

| Interfaz | Descripción |
|---|--|
|  | <ul style="list-style-type: none"> • Esta pantalla permite realizar el diagnóstico de plagas en los cultivos en tiempo real |

Tabla 18
Pantalla Control de Plagas

| Interfaz | Descripción |
|--|---|
|  | <ul style="list-style-type: none"> Esta pantalla muestra las indicaciones que debe realizar el agricultor, si detecta presencia de plagas en sus cultivos. |

Tabla 19
Pantalla Información de Plagas

| Interfaz | Descripción |
|---|--|
|  | <ul style="list-style-type: none"> Esta pantalla muestra la información sobre las plagas mosca minadora y punta morada de la papa, así como algunas recomendaciones sobre su cultivo. |

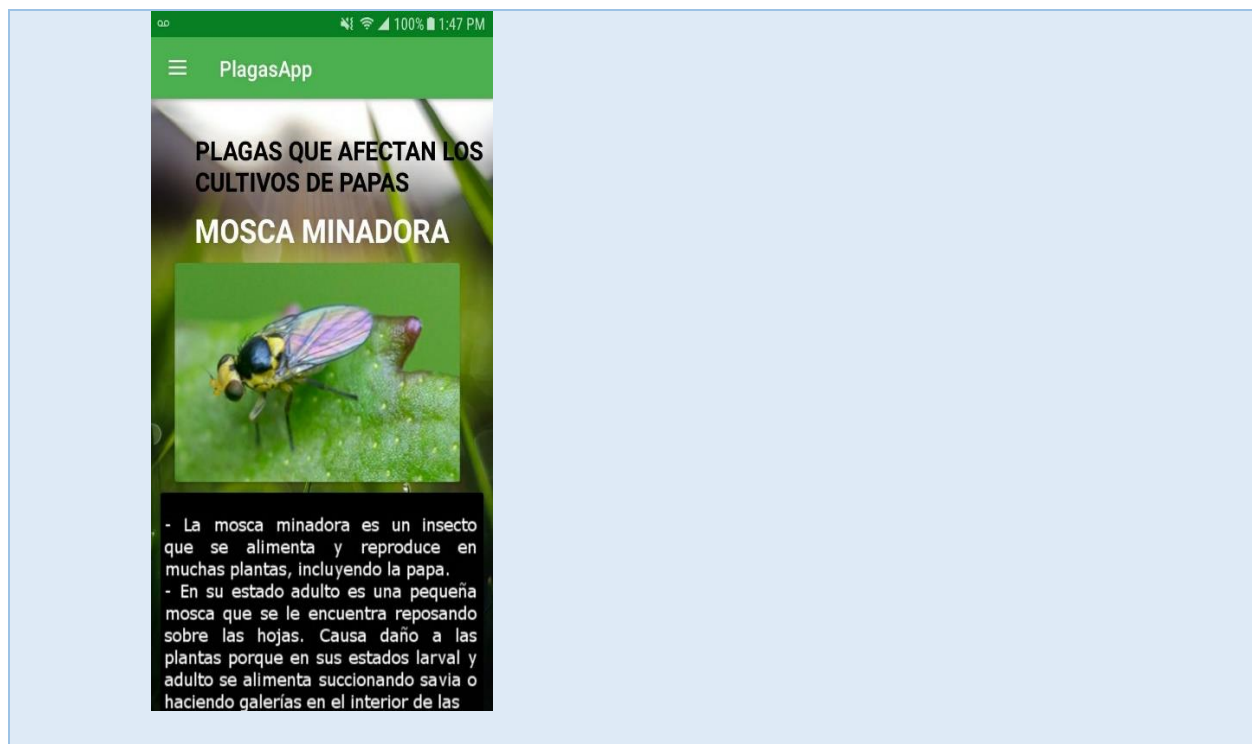


Tabla 20
Pantalla Proceso Diagnóstico Plagas

| Interfaz | Descripción |
|----------|--|
| | <ul style="list-style-type: none"> Esta pantalla permite visualizar información sobre los pasos para realizar el proceso manual de diagnóstico de plagas en cultivos de papas |

3.5.3. Producción, estabilización y pruebas

a. Implementación de la aplicación

Una vez finalizado con el diseño de la aplicación móvil se inicia con la implementación de las funcionalidades de la misma, a través del desarrollo iterativo incremental garantizando la calidad a través de pruebas que permitan validar que la aplicación cumple con todos los requisitos establecidos en el capítulo anterior.

a.1. Estándares de codificación

Variables

Los nombres de las variables estarán definidos la primera palabra en minúscula y el primer carácter de la segunda en mayúscula.

Métodos

Los métodos estarán establecidos de la siguiente manera, varias palabras de un método la primera palabra en minúscula y el primer carácter de la segunda en mayúscula.

Clases

Los nombres de las clases que contengan varias palabras, se utilizara el primer carácter de la primera palabra en mayúscula y el primer carácter de la segunda en mayúscula.

b. Pruebas de la aplicación móvil

Una vez concluida con la fase de desarrollo, empezamos a la creación de las pruebas para determinar la efectiva funcionalidad de la aplicación desarrollada según los requisitos iniciales.

Para esto, realizaremos el siguiente plan de pruebas:

- Pruebas de interfaz
- Pruebas de aceptación

b.1. Pruebas de interfaz

Las pruebas de interfaz permiten evaluar el diseño de la interfaz de la aplicación, asegurando que los elementos como colores, tamaños, letras estén establecidos de manera adecuada.

Analizamos la interfaz principal de la aplicación

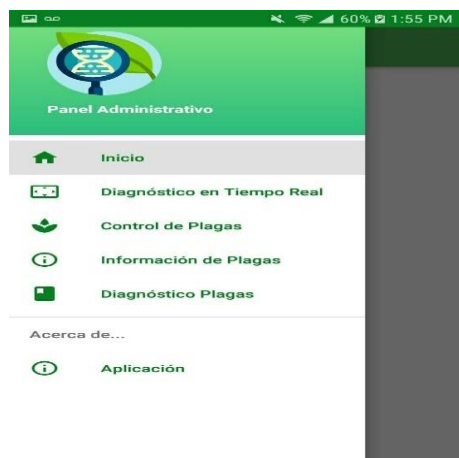


Figura 34. Pantalla de prueba de interfaz de la aplicación

Podemos observar que la interfaz principal, es amigable y de fácil uso para el usuario, siendo la principal funcionalidad la opción de diagnóstico en tiempo real que funciona a través de la activación de la cámara del dispositivo.

b.2. Pruebas de aceptación

Tabla 21

Prueba de aceptación Detectar posibles amenazas de plagas en hojas de cultivos de papas

| Pruebas de aceptación | |
|---|--|
| Caso de prueba: Detectar posibles amenazas de plagas en hojas de cultivos de papas. | Opción de prueba: Detectar posibles amenazas de plagas en hojas de cultivos de papas. |
| Numero de caso de prueba: 1 | Numero de HU:1 |
| Nombre de caso de prueba: Detectar posibles amenazas de plagas en hojas de cultivos de papas con el dispositivo móvil | |
| Precondiciones: Una vez abierta la aplicación seleccionamos del menú la opción “Diagnóstico en Tiempo Real”, esta activará la cámara del dispositivo móvil y mediante el uso del modelo de red neuronal desarrollado en el capítulo 3, se marcará un cuadro sobre la plaga detectada con un porcentaje de precisión. | |
| Resultado esperado: La aplicación se abrirá sin ningún error en el dispositivo móvil. La aplicación detectara posible amenaza de plagas en los cultivos de papas. | |
| Evaluación: La aplicación funcionó correctamente. Se detecto posibles amenazas de plagas en las papas de manera satisfactoria. | |

Tabla 22

Prueba de aceptación Información de las plagas existentes en cultivos de papas

| Pruebas de aceptación | |
|---|---|
| Caso de prueba: Información de las plagas existentes en cultivos de papas. | Opción de prueba: Información de las plagas existentes en cultivos de papas. |
| Numero de caso de prueba: 2 | Numero de HU:2 |
| Nombre de caso de prueba: Mostrar la información de las plagas existentes en cultivos de papas en la aplicación móvil. | |
| Precondiciones: Una vez abierta la aplicación seleccionamos del menú la opción “Información de Plagas”, esta mostrara la información sobre las principales plagas que afectan los cultivos de papas. | |
| Resultado esperado: La aplicación se abrirá sin ningún error en el dispositivo móvil. La aplicación mostrara la información correcta sobre las plagas. | |
| Evaluación: La aplicación mostro la información correcta. | |

Tabla 23

Prueba de aceptación Información de insecticidas existentes para el control de plagas

| Pruebas de aceptación | |
|--|--|
| Caso de prueba: Información de insecticidas existentes para el control de plagas. | Opción de prueba: Información de insecticidas existentes para el control de plagas. |
| Numero de caso de prueba: 3 | Numero de HU:3 |
| Nombre de caso de prueba: Mostrar la Información de insecticidas existentes para el control de plagas en la aplicación móvil. | |
| Precondiciones: Una vez abierta la aplicación seleccionamos del menú la opción “Control de Plagas”, esta mostrara la información sobre los insecticidas existentes para el control de plagas. | |
| Resultado esperado: La aplicación se abrirá sin ningún error en el dispositivo móvil. La aplicación mostrara la información correcta sobre los insecticidas para el control de plagas. | |
| Evaluación: La aplicación mostro la información correcta. | |

Tabla 24

Prueba de aceptación Información sobre el proceso de diagnóstico de plagas en cultivos de papas.

| Pruebas de aceptación | |
|---|--|
| Caso de prueba: Información sobre el proceso de diagnóstico de plagas en cultivos de papas. | Opción de prueba: Información sobre el proceso de diagnóstico de plagas en cultivos de papas. |
| Numero de caso de prueba: 4 | Numero de HU:4 |
| Nombre de caso de prueba: Mostrar la Información sobre el proceso de diagnóstico de plagas en cultivos de papas en la aplicación móvil. | |
| Precondiciones: Una vez abierta la aplicación seleccionamos del menú la opción “Proceso Diagnóstico Plagas”, esta mostrara la información sobre el proceso de diagnóstico de plagas en hojas de cultivos de papas. | |
| Resultado esperado: La aplicación se abrirá sin ningún error en el dispositivo móvil. La aplicación mostrara la información correcta sobre el proceso de diagnóstico de plagas en hojas de cultivos de papas | |
| Evaluación: La aplicación mostro la información correcta. | |

CAPITULO IV

VALIDACIÓN Y ANÁLISIS DE LA APLICACIÓN

4.1. Introducción

En este capítulo se realiza la validación de la aplicación sistema software de aprendizaje profundo que optimice el diagnóstico de plagas en hojas de cultivos de papas, en el sector la libertad parroquia Toacaso provincia de Cotopaxi, en un ambiente controlado, aplicando un conjunto de procedimientos estandarizados de investigación denominado encuesta, esta técnica de investigación descriptiva permite recopilar datos a través del uso de un cuestionario, el mismo que tiene como finalidad extraer información sobre el grado de impacto que genera el sistema software desarrollado.

Para el análisis y obtención de los resultados se trabajó con dos grupos de personas establecidos de la siguiente manera:

- Técnicos agrónomos del ministerio de agricultura y ganadería
- Expertos en cultivos de papas

4.2. Procesamiento de los resultados

A continuación, se detalla la estructura del instrumento elaborado para la recolección de datos:

- Valoración de resultados entre 1 y 5 para cada pregunta.
 - Ocasionalmente (en escasas ocasiones)
 - Ordinariamente (en ocasiones puntuales)

- Frecuentemente (en la mayor parte)
- Muy Frecuentemente (en casi todas las ocasiones)
- Siempre
- Se deberá marcar con una (X) el valor seleccionado.

| | | | | |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|

- Se sumará todas las (X) por cada columna correspondiente.
- El resultado obtenido se multiplicará por el valor de cada columna, de esta forma se obtiene el total del puntaje de la columna.
- Se sumará el total de valor de cada columna y se dividirá para el número total de preguntas multiplicado por el valor más alto de la pregunta.
- Los instrumentos están basados en preguntas cerradas, las cuales permitirán conocer el criterio de los encuestados en base a la aplicación móvil desarrollado.

Para la interpretación de resultados, se ha dividido en cuatro grupos de puntajes la cual abarca el 100% de los datos, con el fin de determinar en qué grupo se califica el sistema software propuesto.

A continuación, se muestra la distribución:

De 0% a 39.99%: El sistema software propuesto (aplicación móvil) no se cumple, solo cumple ciertos aspectos o tiene ciertas deficiencias en algunas funcionalidades y se debe mejorar de manera inmediata.

De 40% a 59.99%: El sistema software propuesto se cumple, pero con algunas deficiencias o tiene un nivel de confianza muy bajo al momento de ejecutar sus funcionalidades, y deben tomar medidas para solucionar las deficiencias encontradas.

De 60% a 79.99%: El sistema software propuesto se cumple y presenta inconvenientes mínimos al momento de ejecutar todas sus funcionalidades, es necesario tener en cuenta los puntos más sobresalientes y los más bajos, de ser posible mejorarlos.

De 80% a 100%: El sistema software propuesto optimiza el proceso de diagnóstico de plagas en cultivos de papas, como una herramienta de ayuda tanto para agricultores como técnicos agrónomos.

Los resultados de aplicar este método se recogen en tablas y se realiza una valoración de cada caso, todo con el fin de documentar y validar todas las funcionalidades, que demuestren que se optimiza el proceso de diagnóstico de plagas en cultivos de papas.

Los resultados obtenidos a través del instrumento aplicado para la valoración de la aplicación, se realizaron a través de la aplicación de preguntas que permiten medir tanto parámetros de usabilidad como de funcionalidad, con los cuales se conoce el punto de vista que tienen las personas encuestadas ante el sistema software propuesto.

4.2.1. Análisis y resultados de la aplicación del instrumento a los técnicos agrónomos del Ministerio de Agricultura y Ganadería (MAG).

En primer lugar, se detalla el análisis e interpretación recopilada del instrumento aplicado a los técnicos agrónomos del ministerio de agricultura y ganadería (MAG) que se muestra en el ANEXO B.

Para esto se contó con la ayuda de diez técnicos agrónomos, para el cálculo de estos valores se dividió el valor total de puntos obtenidos (ST) para el valor que resulta de multiplicar el número total de preguntas de la encuesta por el valor más alto de la misma, y multiplicar por el 100%. Los cálculos se observan en la tabla 24, de igual manera la incorporación de los resultados se muestra en la figura 24.

Tabla 25

Resultados de aceptación de los técnicos agrónomos del ministerio de agricultura y ganadería (MAG)

| Valor | 1 | 2 | 3 | 4 | 5 |
|--|-----------------------------|----|----|-----|-----|
| Total, de (X) | 0 | 3 | 31 | 99 | 47 |
| Multiplicación | *1 | *2 | *3 | *4 | *5 |
| Resultado parcial | 0 | 6 | 93 | 396 | 235 |
| Suma total de puntos obtenidos (ST) | | | | | 730 |
| RESULTADO $[(ST / (180 * 5)) * 100\%]$ | 81,11% de aceptación | | | | |

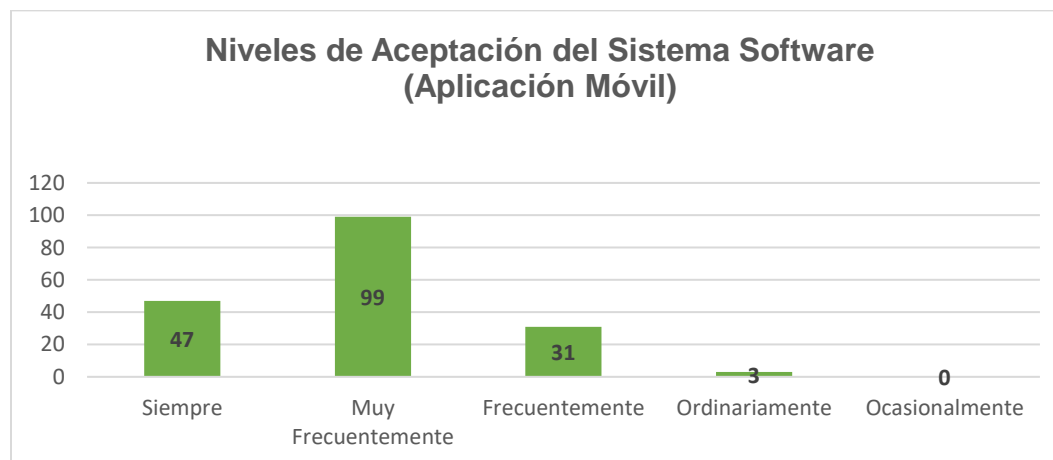


Figura 35. Criterio de aceptación del sistema software (aplicación móvil) por parte de los técnicos agrónomos del ministerio de agricultura y ganadería (MAG)

La **figura 35** muestra los resultados de la valoración por los técnicos agrónomos del ministerio de agricultura y ganadería (MAG), el sistema software consiguió un 81,11% de aceptación.

4.2.2. Análisis y resultados de la aplicación del instrumento a los expertos en cultivos de papas.

Se presenta en la tabla 25 el análisis e interpretación de información recolectada del instrumento en la **Figura 36**, aplicado a cinco expertos en cultivos de papas que se muestra en el anexo B.

Tabla 26

Resultados de aceptación de los expertos en cultivos de papas

| Valor | 1 | 2 | 3 | 4 | 5 |
|--|-----------------------------|----|----|-----|-----|
| Total de (X) | 0 | 0 | 10 | 51 | 29 |
| Multiplicación | *1 | *2 | *3 | *4 | *5 |
| Resultado parcial | 0 | 0 | 30 | 204 | 145 |
| Suma total de puntos obtenidos (ST) | | | | | 379 |
| RESULTADO [(ST/ (90*5)) *100%] | 84,22% de aceptación | | | | |

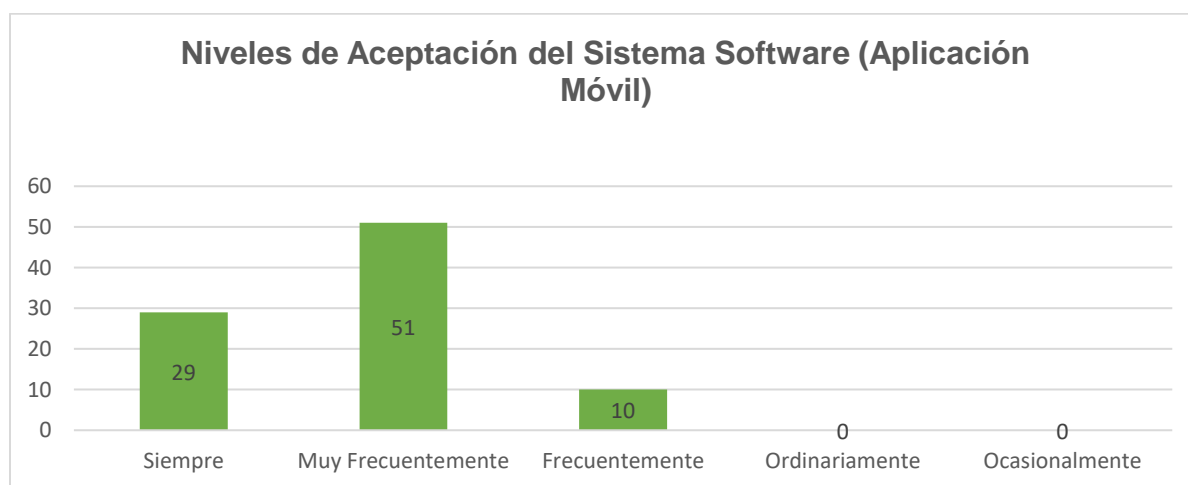


Figura 36. Criterio de aceptación del sistema software (aplicación móvil) por parte de los expertos en cultivos de papas

Los datos de la **Figura 36** se obtuvieron mediante la aplicación del instrumento a cinco expertos en cultivos de papas, como resultado se obtuvo un 84,22% de aceptación del sistema software propuesto.

4.3. Prueba de hipótesis mediante chi cuadrado

Mediante la prueba de hipótesis con chi cuadrado se definirá una comparación entre dos atributos para conocer si existe una relación entre ambos atributos, esto permitirá conocer si una variable depende de la otra para su cumplimiento.

- a) Hipótesis de Investigación:** Si se desarrolla un sistema software de aprendizaje profundo entonces se optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertad parroquia Toacaso Provincia de Cotopaxi
- b) Variable Independiente:** Sistema software basado en aprendizaje profundo (aplicación móvil).
- c) Variable dependiente:** Se optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertad parroquia Toacaso Provincia de Cotopaxi.

4.3.1. Análisis de la hipótesis general

Hipótesis Nula (Ho): El sistema software basado en aprendizaje profundo y la optimización del diagnóstico de plagas en hojas de cultivos de papas son independientes.

Hipótesis Alternativa (Ha): El sistema software basado en aprendizaje profundo y la optimización del diagnóstico de plagas en hojas de cultivos de papas son dependientes.

4.3.2. Cálculo de frecuencias esperadas, correspondientes a cada frecuencia observada.

a. Frecuencia Observada

A continuación, se representan los datos correspondientes a las variables independiente y dependiente de los resultados al aplicar las encuestas a dos grupos. Los datos obtenidos para estas operaciones se encuentran en el anexo C. En la tabla 27 se muestra la valoración de la primera variable, de igual manera en la tabla 28 la valoración de la segunda variable.

Tabla 27

Variable Sistema Software de aprendizaje profundo (aplicación móvil)

| Valoración | Técnicos agrónomos MAG | del | Expertos en cultivos de papas | Total |
|------------------------------|------------------------|-----|-------------------------------|-------|
| 1. Ocasionalmente | 0 | | 0 | 0 |
| 2. Ordinariamente | 3 | | 0 | 3 |
| 3. Frecuentemente | 20 | | 5 | 25 |
| 4. Muy Frecuentemente | 50 | | 28 | 78 |
| 5. Siempre | 17 | | 12 | 29 |

Tabla 28

Variable Optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertada parroquia Toacaso provincia de Cotopaxi

| Valoración | Técnicos agrónomos MAG | del | Expertos en cultivos de papas | Total |
|------------------------------|------------------------|-----|-------------------------------|-------|
| 1. Ocasionalmente | 0 | | 0 | 0 |
| 2. Ordinariamente | 0 | | 0 | 0 |
| 3. Frecuentemente | 11 | | 0 | 11 |
| 4. Muy Frecuentemente | 49 | | 28 | 77 |
| 5. Siempre | 30 | | 17 | 47 |

b) Frecuencia Esperada

En la tabla 29 se muestra la intersección entre los datos de las variables Sistema Software de aprendizaje profundo (aplicación móvil) y Optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertada parroquia Toacaso provincia de Cotopaxi para el cálculo correspondiente.

Tabla 29

Frecuencia Esperada para las Variables Independiente y Dependiente

| | | Sistema Software de aprendizaje profundo (aplicación móvil) | | | | | |
|---|----------------|---|-----|-----|-----|-----|---------|
| | | 1 | 2 | 3 | 4 | 5 | Totales |
| | Valoración | | | | | | |
| Optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertada parroquia Toacaso provincia de Cotopaxi | 1 | 0 | 3 | 25 | 78 | 29 | 135 |
| | 2 | 0 | 3 | 25 | 78 | 29 | 135 |
| | 3 | 11 | 14 | 36 | 89 | 40 | 190 |
| | 4 | 77 | 80 | 102 | 155 | 106 | 520 |
| | 5 | 47 | 50 | 72 | 125 | 76 | 370 |
| | Totales | 135 | 150 | 260 | 525 | 280 | 1350 |

La siguiente ecuación muestra la fórmula para realizar el cálculo de los valores para las variables Sistema Software de aprendizaje profundo (aplicación móvil) frente a Optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertada parroquia Toacaso provincia de Cotopaxi para el cálculo correspondiente.

$$E_{i,j} = \frac{\sum_{i=1}^n O_{i,j} * \sum_{j=1}^n O_{i,j}}{\sum_{i=1}^m \sum_{j=1}^n O_{i,j}}$$

Donde:

m: número de columnas

j: posición de las filas

n: número de filas

O: frecuencia observada

i: posición de las columnas

E: frecuencia esperada

A continuación, mediante el uso de la Ec.1 en la tabla 30 se representa la frecuencia esperada de las dos variables.

Tabla 30

Frecuencia esperada de las dos variables

| | | Sistema Software de aprendizaje profundo (aplicación móvil) | | | | | | |
|---|----------------|---|---------|----------|----------|----------|------|---------|
| | | Valoración | 1 | 2 | 3 | 4 | 5 | Totales |
| Optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertada parroquia Toacaso provincia de Cotopaxi | 1 | 13,5 | 15 | 26 | 52,5 | 28 | 135 | |
| | 2 | 13,5 | 15 | 26 | 52,5 | 28 | 135 | |
| | 3 | 19 | 21,1111 | 36,5926 | 73,8889 | 39,4074 | 190 | |
| | 4 | 52 | 57,7778 | 100,1481 | 202,2222 | 107,8519 | 520 | |
| | 5 | 37 | 41,1111 | 71,2593 | 143,8889 | 76,7407 | 370 | |
| | Totales | 135 | 150 | 260 | 525 | 280 | 1350 | |

4.3.3. Cálculo del valor de chi cuadrado

En esta sección se procede al cálculo de chi cuadrado que permitirá corroborar la dependencia de la variable Optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertada parroquia Toacaso provincia de Cotopaxi para el cálculo correspondiente frente a la variable Sistema Software de aprendizaje profundo (aplicación móvil) comprobando así la hipótesis planteada.

A continuación, en la ecuación se muestra la fórmula para el cálculo del valor de Chi Cuadrado:

$$x^2 = \sum_{i=1}^m \sum_{j=1}^n \frac{(o_{i,j} - E_{i,j})^2}{E_{i,j}}$$

Donde:

x: valor Chi Cuadrado

j: posición filas

m: número de columnas

O: frecuencia observada

n: número de filas

E: frecuencia esperada

i: posición columnas

En la tabla 31 se detalla el cálculo de Chi Cuadrado para ambas variables

Tabla 31

Valor de Chi Cuadrado

| | Valoración | Sistema Software de aprendizaje profundo (aplicación móvil) | | | | | Totales |
|---|----------------|---|---------|--------|---------|--------|-----------------|
| | | 1 | 2 | 3 | 4 | 5 | |
| Optimiza el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertada parroquia Toacaso provincia de Cotopaxi | 1 | 13,5 | 9,6 | 0,0385 | 12,3857 | 0,0357 | 35,5599 |
| | 2 | 13,5 | 9,6 | 0,0385 | 12,3857 | 0,0357 | 35,5599 |
| | 3 | 3,3684 | 2,3953 | 0,0096 | 3,0904 | 0,0089 | 8,8726 |
| | 4 | 12,0192 | 8,5470 | 0,0342 | 11,0272 | 0,0318 | 31,6594 |
| | 5 | 2,7027 | 1,9219 | 0,0077 | 2,4796 | 0,0072 | 7,1191 |
| | Totales | 45,0903 | 32,0642 | 0,1285 | 41,3687 | 0,1193 | 118,7710 |

De la tabla 29 se observa que el resultado para el valor de Chi Cuadrado es:

$$\chi^2_{\text{observado}} = 118,7710$$

4.3.4. Cálculo del valor crítico de chi cuadrado

Para obtener el valor crítico de Chi Cuadrado, se realiza el cálculo de los grados de libertad para el Chi Crítico con un nivel de significancia supuesto, luego se procede a observar en la tabla de distribución de Chi crítico para encontrar su valor correspondiente.

Nivel de Significancia

$$\alpha = 0,05$$

Grados de Libertad (gl.)

$$gl. = (\text{numero columnas} - 1) * (\text{numero filas} - 1)$$

$$gl. = (5 - 1)(5 - 1)$$

$$gl. = 4 * 4$$

$$gl. = 16$$

En la tabla 32 se busca el valor del Chi crítico según el Nivel de Significancia (filas) y los Grados de Libertad (columnas) obtenidos anteriormente.

Tabla 32
Distribución Chi Cuadrado Crítico

| Grados de Libertad (gl.) | Áreas de Extremo Superior (α) | | | | | |
|--------------------------|--|--------|--------|--------|--------|--------|
| | 0.25 | 0.10 | 0.05 | 0.025 | 0.01 | 0.005 |
| 1 | 1.323 | 2.706 | 3.841 | 5.024 | 6.635 | 7.879 |
| 2 | 2.773 | 4.605 | 5.991 | 7.378 | 9.210 | 10.597 |
| 3 | 4.108 | 6.251 | 7.815 | 9.348 | 11.345 | 12.838 |
| 4 | 5.385 | 7.779 | 9.488 | 11.143 | 13.277 | 14.860 |
| 5 | 6.662 | 9.236 | 11.071 | 12.883 | 15.086 | 16.750 |
| 6 | 7.841 | 10.645 | 12.592 | 14.449 | 16.812 | 18.548 |
| 7 | 9.037 | 12.017 | 14.067 | 16.013 | 18.475 | 20.278 |
| 8 | 10.129 | 13.362 | 15.507 | 17.535 | 20.090 | 21.955 |
| 9 | 11.389 | 14.684 | 16.919 | 19.023 | 21.666 | 23.589 |
| 10 | 12.549 | 15.987 | 18.307 | 20.483 | 23.209 | 25.188 |
| 11 | 13.701 | 17.275 | 19.675 | 21.920 | 24.725 | 26.757 |
| 12 | 14.845 | 18.549 | 21.026 | 23.337 | 26.217 | 28.299 |
| 13 | 15.984 | 19.812 | 22.362 | 24.736 | 27.688 | 29.819 |
| 14 | 17.117 | 21.064 | 23.685 | 26.119 | 29.141 | 31.319 |
| 15 | 18.245 | 22.307 | 24.996 | 27.488 | 30.578 | 32.801 |
| 16 | 19.369 | 23.542 | 26.296 | 28.845 | 32.000 | 34.267 |
| 17 | 20.489 | 24.769 | 27.587 | 30.191 | 33.409 | 35.718 |

Chi Cuadrado Crítico: $\chi^2_{\text{critico}} = 26.296$

4.3.5. Comparación entre el valor esperado y el valor crítico

Una vez calculado el valor observado y valor crítico de Chi Cuadrado se procede a comparar estos valores y aplicar la regla de decisión para corroborar el cumplimiento de la hipótesis planteada.

En la **Figura 37** se observa que el valor de Chi Cuadrado se encuentra en la zona de rechazo

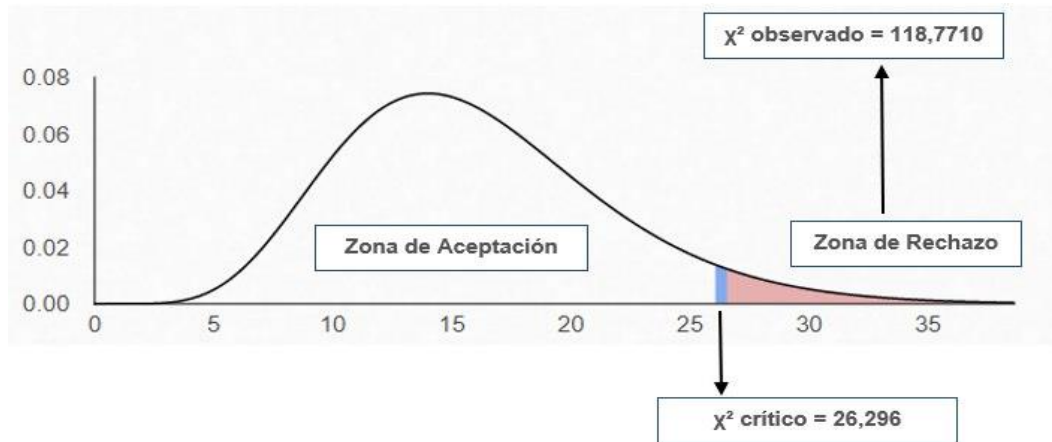


Figura 37. Prueba de Chi Cuadrado Dependiente

Regla de decisión:

- Se acepta la hipótesis nula (H_0), si: $x^2_{observado} < x^2_{crítico}$
- Se acepta la hipótesis alternativa (H_A), si: $x^2_{observado} > x^2_{crítico}$

En nuestro caso tenemos los siguientes valores:

$$x^2_{observado} = 118,7710 \text{ y } x^2_{crítico} = 26,296$$

Entonces se puede ver el valor de Chi Cuadrado observado es mayor que el valor crítico ($118,7710 > 26,296$) por lo tanto se acepta la hipótesis Alternativa (H_A).

4.3.6. Conclusiones de los resultados presentados

El análisis de resultados con la prueba de Chi Cuadrado realizada para cada indicador del proyecto permite corroborar que, la optimización del diagnóstico de plagas en hojas de cultivos de papas sector la Libertad parroquia Toacaso provincia de Cotopaxi depende

claramente del sistema software (aplicación móvil) basado en aprendizaje profundo, retornando así los resultados esperados.

Se pudo observar que según los datos de las encuestas aplicadas indican que el sistema software (aplicación móvil) de aprendizaje profundo, es apto para su uso, así cumpliendo con la hipótesis planteada en el proyecto de titulación, la cual hace mención a la optimización del diagnóstico de plagas en hojas de cultivo de papas en el sector la Libertad parroquia Toacaso provincia de Cotopaxi.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Introducción

En el presente capítulo se muestra las conclusiones y recomendaciones obtenidas en el desarrollo de la investigación.

5.2. Conclusiones

- Se logro cumplir con el objetivo del proyecto, desarrollar un sistema software de aprendizaje profundo que optimice el diagnóstico de plagas en hojas de cultivos de papas en el sector la Libertad parroquia Toacaso provincia de Cotopaxi.
- La elaboración del marco teórico permitió obtener información con respecto al método y técnicas en el proceso de diagnóstico de plagas en cultivos de papas, también conocer los diferentes métodos que existen en el campo del aprendizaje profundo, con la finalidad de desarrollar un sistema software útil para el diagnóstico de pagas en cultivos de papas.
- El estudio comparativo realizado entre las diferentes arquitecturas de redes neuronales existentes en el campo del aprendizaje profundo, se pudo conocer que la arquitectura MobileNet es la más adecuada, debido a que son redes neuronales especializadas en el procesamiento de imágenes y están enfocados para el desarrollo de aplicaciones móviles.
- La construcción del dataset y el etiquetado de imágenes se realizó de forma manual, se realizó un tratamiento de las imágenes las cuales fueron clasificadas,

procesadas y se realizó un aumento de datos para completar un total de 20000 imágenes que fueron utilizadas para el entrenamiento de la red neuronal, se considera a esta etapa como la que tomo más tiempo realizar para el desarrollo del proyecto de investigación.

- El entrenamiento de la red neuronal se realizó mediante aprendizaje de transferencia, debido a que esta técnica permite acelerar el proceso de entrenamiento de días a horas y de igual manera el conjunto de datos (imágenes) disminuye de millones a miles.
- El desarrollo de la aplicación móvil se realizó a través del análisis de requerimientos de la aplicación basado en la metodología Mobile-D, la cual permite crear un entorno de trabajo estable y obtener resultados funcionales en cada iteración realizado.
- Se determinó que el uso de la tecnología es una opción útil para ayudar a agricultores y técnicos agrónomos en el proceso de detección de plagas en sus cultivos de papas, al ser una herramienta que logro llamar la atención por todas las funcionalidades e información que brinda y la facilidad de uso en sectores rurales, ya que no es necesario una conexión a internet para el funcionamiento de la aplicación.
- La validación de la aplicación móvil se realizó mediante un instrumento de encuesta realizada a dos grupos de personas, técnicos agrónomos del ministerio de agricultura y ganadería (MAG) y expertos en cultivos de papas, lo cual permitió validar las funcionalidades y la información que proporciona la aplicación,

observando un grado de aceptación de 81, 11% y 84.22% respectivamente con relación a los indicadores establecidos.

- La aplicación permite la detección de plagas en tiempo real a través de la cámara del dispositivo móvil, una vez detectada la presencia de plagas en sus cultivos, puede acceder a la opción control de plagas donde puede encontrar diferentes tipos de control dependiendo de la época en la que se encuentre el cultivo, de igual manera puede encontrar información relevante sobre las principales plagas que actualmente están afectando mayormente a los cultivos y la información enfocada a los agricultores para realizar un proceso de diagnóstico de plagas a sus cultivos.

5.3. Recomendaciones

- Para el Desarrollo de aplicaciones móviles, se recomienda utilizar la metodología Mobile-D debido a que facilita ciclos de desarrollo rápidos en grupos pequeños, también detecta errores tempranos para ser solventados en el tiempo esperado y de esa manera crear aplicaciones móviles de calidad.
- Se recomienda tener un periodo de adaptación tanto en técnicos agrónomos como en agricultores en el uso del aplicativo software, debido a que los mismos no cuentan con poco a nada de experiencia, por lo que se supone un cambio para ellos. Por tal motivo, es necesario brindarles un tiempo para que ellos se puedan adaptar al aplicativo software.
- Se recomienda que la aplicación desarrollada puede ser utilizada por agricultores y técnicos agrónomos con el fin de optimizar el diagnóstico de plagas, poder aplicar

un control de las plagas encontradas y de esta manera poder disminuir las pérdidas que se tienen por la presencia de plagas en sus cultivos.

- Para tener un buen rendimiento de la aplicación móvil se recomienda el uso de un dispositivo con versión mínima de Android 7.0, y cámaras de capacidad mínima de 13 megapíxeles.
- Disponer de una computadora con excelentes recursos de procesador y tarjeta gráfica que permita realizar el entrenamiento de la red neuronal de manera rápida.
- Se recomienda investigar el uso de otros modelos de red neuronal, realizar una comparación entre varios modelos y ver cuál es el más factibles para su aplicación en campos agrícolas.
- Investigar sobre una herramienta que permita realizar el etiquetado de imágenes de forma automática, de ser posible desarrollar una ya que el etiquetado manual requiere un lapso de tiempo más extenso.
- Se recomienda a futuro incrementar funcionalidades en la aplicación como notificaciones, sistema de base de datos y geolocalización.
- Se recomienda a futuro incrementar el número de plagas a ser detectadas y la automatización en el diagnóstico de plagas en otros tipos de cultivos

REFERENCIAS BIBLIOGRÁFICAS

- Agencia de Regulación y Control Fito y Zoosanitario. (11 de Julio de 2019). *Agrocalidad, Mision y Vision*. Recuperado el 19 de Septiembre de 2019, de Agrocalidad, Mision y Vision: <http://www.agrocalidad.gob.ec/mision-vision/>
- Ajeet, R., Manjusha, P., & Siddharth, R. (2018). Application of Deep Learning for Object Detection. *International Conference on Computational Intelligence and Data Science (ICCIDS 2018)*, 3-4. Recuperado el 22 de Noviembre de 2019
- Álban, E., & Bassante, A. (2015). Desarrollo de un aplicación móvil de orinetación y ubicación para personas que ingresen en la Universidad de las Fuerzas Armadas Espe Extensión Latacunga Campus Centro. Latacunga, Cotopaxi, Ecuador. Recuperado el 25 de Noviembre de 2019
- Amorin, C. (18 de abril de 2017). *Red Neuronal en Python con numpy*. Recuperado el 26 de Noviembre de 2019, de Red Neuronal en Python con numpy: <https://artfromcode.wordpress.com/2017/04/18/red-neuronal-en-python-con-numpy-parte-1/>
- Andrade Vera, J. A. (11 de 09 de 2018). Aplicación móvil para la detección y tratamiento de daños de los cultivos de la parroquia Taura del cantón Durán, mediante el uso de software de análisis de imagen basado en técnicas de machine learning. Duran, Guayas, Ecuador. Recuperado el 26 de Noviembre de 2019
- Aprende Machine Learning. (5 de Febrero de 2019). *Machine Learning en la Nube? Google Colaboratory con GPU!* Recuperado el 5 de Noviembre de 2019, de

Machine Learning en la Nube? Google Colaboratory con GPU:
<https://www.aprendemachinellearning.com/machine-learning-en-la-nube-google-colaboratory-con-gpu/>

Arenas Hernández, L. (s.f.). *PROGRAMACION ORIENTADA A OBJETOS EN JAVA*. Mexico. Recuperado el 29 de Noviembre de 2019

Ávila, R., Eduardo, P., Orozco, L. D., Sánchez, L. M., & Hidalgo, M. (2018). Deep Learning, una revisión. 5-7. Recuperado el 28 de Noviembre de 2019

Balaguera, Y. (2013). Metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles. Estado actual. *Journal Technology*, 8-12. Recuperado el 28 de Noviembre de 2019

Bayer CropScience. (2019). *Moscas Minadoras*. Recuperado el 7 de Noviembre de 2019, de Moscas Minadoras: <https://cropscience.bayer.com.ar/content/moscas-minadoras>

Bou, A., Shahin, I., Attili, I., & Shaalan, K. (1 de Febrero de 2019). *IEEE*. Recuperado el 30 de Noviembre de 2019, de Reconocimiento de voz usando redes neuronales profundas: una revisión sistemática: <https://ieeexplore.ieee.org/document/8632885>

Buhigas, J. (14 de Febrero de 2018). *Todo lo que necesitas saber sobre TensorFlow, la plataforma para Inteligencia Artificial de Google*. Recuperado el 30 de Noviembre de 2019, de Todo lo que necesitas saber sobre TensorFlow, la plataforma para Inteligencia Artificial de Google: <https://puentesdigitales.com/2018/02/14/todo-lo->

que-necesitas-saber-sobre-tensorflow-la-plataforma-para-inteligencia-artificial-de-google/

Cáceres, C., Amaya, D., & Ramos, O. (2015). Methodology for pest damage recognition in *Begonia semperflorens* link & Otto (sugar flower) crop through image processing. *Plant and Crop Protection / Sanidad Vegetal y Producción de Cultivos*, 257-264. Recuperado el 29 de Noviembre de 2019

Cevallos, G., & Mantilla, J. (2008). Levantamiento de Plagas Insectiles de Papa (*Solanum tuberosum*) en cuatro formaciones ecológicas de la serranía ecuatoriana. Sangolqui, Pichicnha, Ecuador. Recuperado el 1 de Diciembre de 2019

Cisneros, F. (2010). *EL MANEJO INTEGRADO DE PLAGAS*. Recuperado el 1 de Diciembre de 2019

DIGITAL TECH INSTITUTE. (9 de Abril de 2018). *LOS 7 MEJORES LENGUAJES DE PROGRAMACIÓN PARA IA*. Obtenido de LOS 7 MEJORES LENGUAJES DE PROGRAMACIÓN PARA IA: <https://www.digitaltechinstitute.com/8-mejores-lenguajes-de-programacion-para-ia/>

Dot CSV. (25 de Octubre de 2019). ¿Qué es el Machine Learning? ¿Y Deep Learning? Un mapa conceptual. *¿Qué es el Machine Learning? ¿Y Deep Learning? Un mapa conceptual*. España. Recuperado el 02 de Octubre de 2019

Espinoza Mendoza, L. E. (2018). *TENSORFLOW COMO ALTERNATIVA A HERRAMIENTAS ESTADISTICAS*. 6-7. Recuperado el 1 de Diciembre de 2019

Ferentinos, K. (2018). Deep learning models for plant disease detection and diagnosis.

Computers and Electronics in Agriculture, 8. Recuperado el 5 de Diciembre de 2019

Food and Agriculture Organization of the United Nations. (6 de Junio de 2017). *Agencia*

Ecuatoriana de Aseguramiento de la Calidad del Agro-AGROCALIDAD.

Recuperado el 08 de Septiembre de 2019, de Agencia Ecuatoriana de

Aseguramiento de la Calidad del Agro-AGROCALIDAD:

<https://www.ippc.int/es/countries/ecuador/reportingobligation/2015/10/agencia-ecuatoriana-de-aseguramiento-de-la-calidad-del-agro-agrocalidad/>

Gago Utrera, R. (Junio de 2017). Uso de algoritmos de aprendizaje automático aplicados

a bases de datos genéticos. Catalunya, España. Recuperado el 6 de Diciembre de

2019

Gallardo, G., & Mantilla, J. (Noviembre de 2008). LEVANTAMIENTO DE PLAGAS

INSECTILES DE PAPA. Quito, Pichincha, Ecuador. Recuperado el 6 de Diciembre

de 2019

Gobierno Autónomo Descentralizado Parroquia Rural de Toacaso. (2019).

DIAGNOSTICO DE LA PARROQUIA TOACASO. Latacunga. Recuperado el 10 de

Octubre de 2019

Gobierno Autónomo Descentralizado Parroquia Rural de Toacaso. (2019). *Situacion*

Geográfica. Recuperado el 10 de Noviembre de 2019, de Situacion Geográfica:

<http://toacaso.gob.ec/cotopaxi/?p=69>

González, A. (2018). Aplicaciones de técnicas de inteligencia artificial basadas en aprendizaje profundo al análisis y mejora de la eficiencia de procesos industriales. 26-29. Recuperado el 10 de Diciembre de 2019

Hopkins, M. (08 de Septiembre de 2017). *AgriAI: Deep Learning in Indian Agriculture*. Recuperado el 15 de Diciembre de 2019, de AgriAI: Deep Learning in Indian Agriculture: <https://www.precisionag.com/market-watch/agriai-deep-learning-in-indian-agriculture/>

Howard, A. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision. 1-4. Recuperado el 16 de Diciembre de 2019

Hsiao, E. (20 de Abril de 2018). *Community Spotlight — Nuru, a mobile app by PlantVillage to detect crop disease in Africa*. Recuperado el 17 de Diciembre de 2019, de Community Spotlight — Nuru, a mobile app by PlantVillage to detect crop disease in Africa: <https://heartbeat.fritz.ai/community-spotlight-nuru-a-mobile-app-by-plantvillage-to-detect-crop-disease-in-africa-28d142bf63d5>

INIAP. (2017). *Ficha Técnica de la Punta Morada*. Quito.

Instituto Nacional de Investigaciones Agropecuarias. (1991). *Aspectos tecnológicos del cultivo de papa en el Ecuador*. Quito.

Instituto Nacional de Estadística y Censos. (2017). *Ficha de cultivo de papa*. Cotopaxi. Recuperado el 18 de Octubre de 2019

Instituto Nacional de Investigaciones Agropecuarias. (1976). *Tercer Curso Nacional Sobre Tecnología de cultivo de Papa*. Quito. Recuperado el 18 de Noviembre de 2019

Instituto Nacional de Investigaciones Agropecuarias. (1986). *Quinto curso sobre producción de semilla de papa a partir de cultivo de tejidos, mejoramiento y tecnología de cultivo*. Quito.

Instituto Nacional de Investigaciones Agropecuarias. (2002). *Cultivo de papa en el Ecuador*. Quito. Recuperado el 03 de Octubre de 2019

Instituto Nacional de Investigaciones Agropecuarias. (2002). *El cultivo de papa en el Ecuador*. Quito. Recuperado el 15 de Octubre de 2019

Instituto Nacional de Investigaciones Agropecuarias. (2013). *El cultivo de papa*. Quito. Recuperado el 12 de Octubre de 2019

Instituto Nacional de Investigaciones Agropecuarias. (2013). *Guía Fotográfica de las principales plagas del cultivo de papa en el Ecuador*. Quito. Recuperado el 16 de Octubre de 2019

Instituto Nacional de Investigaciones Agropecuarias. (2018). *INIAP ejecuta un plan emergente frente a la presencia de Punta Morada de la Papa en Ecuador*. Quito. Recuperado el 18 de Octubre de 2019

Instituto Nacional de Investigaciones Agropecuarias. (08 de Agosto de 2019). *INIAP inaugura invernadero automatizado de producción de semilla básica de papa*. Recuperado el 20 de Octubre de 2019, de INIAP inaugura invernadero

automatizado de producción de semilla básica de papa:
<http://www.iniap.gob.ec/pruebav3/iniap-inaugura-invernadero-automatizado-de-produccion-de-semilla-basica-de-papa/>

Instituto Nacional de Investigaciones Agropecuarias. (11 de Julio de 2019). *Instituto Nacional de Investigaciones Agropecuarias - La Institución*. Recuperado el 21 de Octubre de 2019, de Instituto Nacional de Investigaciones Agropecuarias - La Institución: <http://www.iniap.gob.ec/pruebav3/intitucion/>

Ivette, A., Rodrigo, B., María, P. C., Pamela, T., Camila, S., Mincy, V., & Sandra, M. (01 de 04 de 2015). *Manual Interactivo de la papa INIA*. Recuperado el 18 de Diciembre de 2019, de Manual Interactivo de la papa INIA: <https://manualinia.papachile.cl/?page=consumo&etapa=7>

Java. (7 de Septiembre de 2016). *Conozca más sobre la tecnología Java*. Recuperado el 02 de Octubre de 2019, de Conozca más sobre la tecnología Java: <https://www.java.com/es/about/>

Kamilaris, A. (2018). Deep learning in agriculture : A survey. *Computers and Electronics in Agriculture*, 70-90. Recuperado el 22 de Diciembre de 2019

Konstantinos, P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture*, 311-318. Recuperado el 23 de Diciembre de 2019

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Macmillan Publishers Limited*, 1-9. Recuperado el 24 de Diciembre de 2019

Linga, S. (28 de Octubre de 2018). *Introducción a Google Colab*. Recuperado el 24 de Diciembre de 2019, de Introducción a Google Colab: https://shranith.github.io/2018/10/28/google_colab_intro.html

M Manoj, k., Neelima, Harshali, & M Venu, G. R. (2018). Image classification using Deep learning. *International Journal of Engineering & Technology*, 1-2. Recuperado el 25 de Diciembre de 2019

MathWorks. (2019). *Deep Learning*. Recuperado el 5 de Octubre de 2019, de Deep Learning: <https://es.mathworks.com/discovery/deep-learning.html>

Méndez, P., & Inostroza, J. (2009). *Manual de papa para la Araucanía: Manejo de cultivo, enfermedades y almacenaje*. Recuperado el 25 de Diciembre de 2019

Ministerio de Agricultura y Ganadería. (07 de Agosto de 2019). *En Cotopaxi se fortalece el cultivo de papa*. Recuperado el 15 de Octubre de 2019, de En Cotopaxi se fortalece el cultivo de papa: <https://www.agricultura.gob.ec/en-cotopaxi-se-fortalece-el-cultivo-de-papa/>

Ministerio de Agricultura y Ganadería. (11 de Julio de 2019). *Valores, Mision, Vision del Ministerio de Agricultura y Ganadería*. Recuperado el 20 de Noviembre de 2019, de Valores, Mision, Vision del Ministerio de Agricultura y Ganadería: <https://www.agricultura.gob.ec/valores-mision-vision/>

Mullis, A. (11 de Noviembre de 2017). *Tutorial de Android Studio para principiantes*. Recuperado el 26 de Diciembre de 2019, de Tutorial de Android Studio para

principiantes: <https://www.androidauthority.com/android-studio-tutorial-beginners-637572/>

Nombela Escobar, B. (07 de 2011). Aplicación de técnicas de aprendizaje automático para la extracción de información en textos farmacológicos. Madrid, España. Recuperado el 28 de Diciembre de 2019

Pérez, C. (2016). Emotions Recognition using Deep Learning. 12-13. Recuperado el 29 de Diciembre de 2019

Platzi. (2017). *¿Qué es Deep Learning?* Recuperado el 14 de Octubre de 2019, de *¿Qué es Deep Learning?*: <https://platzi.com/blog/que-es-deep-learning/>

Quiza, J. (11 de Mayo de 2018). *Deep Learning Inception-v3 con PySpark*. Recuperado el 30 de Diciembre de 2019, de *Deep Learning Inception-v3 con PySpark*: <https://medium.com/datos-y-ciencia/deep-learning-inception-v3-con-pyspark-b97678c9616b>

Ripa, R., Larral, P., & Rodríguez, s. (2008). *Manejo integrado de plagas (MIP)*. Recuperado el 31 de Diciembre de 2019

Robinson, S., Chowdhery, A., & Huang, J. (13 de Julio de 2018). *Capacitación y servicio de un detector de objetos móviles en tiempo real en 30 minutos con Cloud TPU*. Recuperado el 01 de Enero de 2020, de *Capacitación y servicio de un detector de objetos móviles en tiempo real en 30 minutos con Cloud TPU*: <https://medium.com/tensorflow/training-and-serving-a-realtime-mobile-object-detector-in-30-minutes-with-cloud-tpus-b78971cf1193>

Rodríguez, J. (17 de Octubre de 2018). *Cómo empezar con Reconocimiento de Imagen y Redes Neuronales Convolucionales en 5 minutos*. Recuperado el 02 de Enero de 2020, de Cómo empezar con Reconocimiento de Imagen y Redes Neuronales Convolucionales en 5 minutos: <https://planetachatbot.com/c%C3%B3mo-empezar-con-reconocimiento-de-imagen-y-redes-neuronales-convolucionales-en-5-minutos-7f651054dfd7>

Rodríguez, M., Vásquez, M., & Araya, A. (1995). *Guía de identificación y manejo integrado de plagas y enfermedades en piña*. Banacol. Recuperado el 3 de Enero de 2020

Rubi, C. R. (2015). Speech Recognition with Deep. *International Journal of Computer Science and Mobile Computing*, 6-7. Recuperado el 3 de Enero de 2020

Samiran, S. (14 de Junio de 2018). *Deep Learning in Agriculture*. Recuperado el 5 de Enero de 2020, de Deep Learning in Agriculture: <https://medium.com/@farmatronix/deep-learning-in-agriculture-7da1c206434f>

Sánchez, J. M. (2016). SISTEMA DE MONITOREO AGRÍCOLA CON TECNOLOGÍA INALÁMBRICA Y GENERACIÓN DE ALERTAS PARA LA PREVENCIÓN TEMPRANA DE PLAGAS Y ENFERMEDADES EN EL CULTIVO DE PAPA EN LA PARROQUIA QUIMIAG DEL CANTÓN RIOBAMBA DE LA PROVINCIA DE CHIMBORAZO. Riobamaba, Chimborazo, Ecuador. Recuperado el 6 de Enero de 2020

Sumit, S. (15 de diciembre de 2018). *A Comprehensive Guide to Convolutional Neural Networks—the ELI5 way*. Recuperado el 7 de Enero de 2020, de A Comprehensive Guide to Convolutional Neural Networks—the ELI5 way: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>

TensorFlow. (8 de Julio de 2019). *Aprende sobre la relación entre TensorFlow y Keras*. Recuperado el 24 de Septiembre de 2019, de Aprende sobre la relación entre TensorFlow y Keras: <https://www.tensorflow.org/overview>

TensorFlow. (7 de Julio de 2019). *Guía TensorFlow Lite*. Recuperado el 17 de Octubre de 2019, de Guía TensorFlow Lite: https://www.tensorflow.org/lite/guide?hl=es_419

TensorFlow. (14 de 01 de 2019). *TensorBoard: kit de herramientas de visualización de TensorFlow*. Recuperado el 12 de Noviembre de 2019, de <https://www.tensorflow.org/tensorboard>

TensorFlow. (2019). *Tensorflow detection model zoo*. Recuperado el 15 de Noviembre de 2019, de Tensorflow detection model zoo: https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md

Torres, G. (11 de Septiembre de 2018). DESARROLLO DE UNA APLICACIÓN WEB Y MÓVIL BASADA EN MACHINE LEARNING, PARA MONITOREO, REGISTRO Y

CONTROL DE TEMPERATURA Y HUMEDAD EN EL CULTIVO DE LA PITAHAYA. Guayaquil, Guayas, Ecuador. Recuperado el 9 de Enero de 2020

Torres, J. (19 de Abril de 2016). Uso y adaptación de un sistema de Deep Learning para análisis de imágenes en un Supercomputador. Catalunya, España. Recuperado el 12 de Enero de 2020

UNIVERSIDAD DE EL SALVADOR. (2001). *MANUAL TECNICO MANEJO INTEGRADO DE PLAGAS*. Salvador. Recuperado el 6 de Noviembre de 2019

Utrera, J. (5 de Diciembre de 2018). *Deep Learning básico con Keras (Parte 4): ResNet*. Recuperado el 15 de Enero de 2020, de Deep Learning básico con Keras (Parte 4): ResNet: <https://enmilocalfunciona.io/deep-learning-basico-con-keras-parte-4-resnet/>

van Rossum, G. (2017). El Tutorial de Python. En G. van Rossum, *El Tutorial de Python* (págs. 1-3). Recuperado el 12 de Enero de 2020

Vásquez, J. (2014). RED NEURONAL FEEDFORWARD COMO ESTIMADOR DE PATRONES DE CORRINETES EN EL INTERIOR DEL PUERTO DE MANZANILLO SUJETO A LA ACCION DE TSUNAMIS. 5-7. Recuperado el 17 de Enero de 2020

VELASCO ZAPATA, J. A. (31 de Mayo de 2019). DISEÑO Y DESARROLLO DE UN SISTEMA PROTOTIPO DE DIAGNÓSTICO DE AFECCIONES EN PLANTAS. *DISEÑO Y DESARROLLO DE UN SISTEMA PROTOTIPO DE DIAGNÓSTICO DE*

AFECCIONES EN PLANTAS. Cali, Cali, Colombia. Recuperado el 17 de Enero de 2020

Yupangui, L. (2016). VALIDACIÓN DE LOS PARÁMETROS DE CONTROL INTERNO DE CALIDAD EN LA PRODUCCIÓN DE TUBÉRCULO-SEMILLA, EN LA VARIEDAD INIAPLIBERTAD. CADER. Quito, Pichcincha, Ecuador. Recuperado el 18 de Enero de 2020

Zhu, N., Liu, X., Liu, Z., Hu, K., Wang, Y., Tan, J., . . . Guo, Y. (2018). Deep learning for smart agriculture : Concepts , tools , applications , and opportunities. *Int J Agric & Biol Eng Open*, 32-44. Recuperado el 22 de Enero de 2020

ANEXOS



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN SOFTWARE

CERTIFICACIÓN

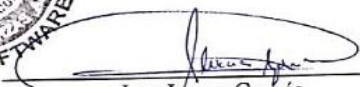
Se certifica que el presente trabajo fue desarrollado por los señores: Choloquina Cocha, Luis Gerardo y Reyes Negrete, Alex Omar como requerimiento parcial a la obtención del título de ingeniería en Software

En la ciudad de Latacunga, miércoles 15 de enero del 2020

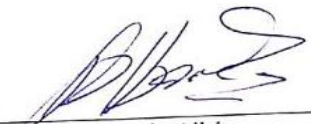
Aprobado por:


Ing. Lucas Garcés
DIRECTOR DEL PROYECTO




Ing. Lucas Garcés
DIRECTOR DE CARRERA




Dr. Darwin Albán
SECRETARIO ACADÉMICO