

UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERIA EN CIENCIAS APLICADAS



**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN**

“SISTEMA DE MONITOREO DE CONDICIONES ÓPTIMAS DE CRECIMIENTO DE
CULTIVOS EN SISTEMAS MICROFARMS POR MEDIO DE VISIÓN POR
COMPUTADOR Y ALGORITMOS DE APRENDIZAJE DE MÁQUINA”

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN ELECTRÓNICA Y REDES DE COMUNICACIÓN**

AUTOR: JORGE ANDRES CRIOLLO NAVARRETE

DIRECTOR: MSC. LUIS EDILBERTO SUAREZ ZAMBRANO

Ibarra – Ecuador

2023



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	1003600390		
APELLIDOS Y NOMBRES:	Jorge Andrés Criollo Navarrete		
DIRECCIÓN:	Calixto Miranda y José Miguel Leoro		
EMAIL:	jacriollon@utn.edu.ec		
TELÉFONO FIJO:		TELÉFONO MÓVIL:	0983792173

DATOS DE LA OBRA	
TÍTULO:	SISTEMA DE MONITOREO DE CONDICIONES ÓPTIMAS DE CRECIMIENTO DE CULTIVOS EN SISTEMAS MICROFARMS POR MEDIO DE VISIÓN POR COMPUTADOR Y ALGORITMOS DE APRENDIZAJE DE MÁQUINA
AUTOR (ES):	Jorge Andrés Criollo Navarrete
FECHA: DD/MM/AAAA	2 de Agosto de 2023
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	☒ PREGRADO ☐ POSGRADO
TÍTULO POR EL QUE OPTA:	Ingeniero en Electronica y Redes de Comunicación
ASESOR /DIRECTOR:	MSC. Luis Suarez

2. CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de esta y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 2 días del mes de Agosto de 2023.



EL AUTOR:

Jorge Criollo



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN:

MAGISTER LUIS SUAREZ, DIRECTOR DEL PRESENTE TRABAJO DE TITULACIÓN CERTIFICA:

Que, el presente trabajo de titulación "SISTEMA DE MONITOREO DE CONDICIONES ÓPTIMAS DE CRECIMIENTO DE CULTIVOS EN SISTEMAS MICROFARMS POR MEDIO DE VISIÓN POR COMPUTADOR Y ALGORITMOS DE APRENDIZAJE DE MÁQUINA" ha sido desarrollado por el señor Jorge Andrés Criollo Navarrete bajo mi supervisión.

Es todo en cuanto puedo certificar en honor de la verdad.

A handwritten signature in black ink, appearing to read "Luis Suarez", enclosed within a circular stamp.

Ing. Luis Edilberto Suarez Zambrano, MsC.

DIRECTOR



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DEDICATORIA

El presente trabajo de titulación agradezco primordialmente a mi madre quien me apoyo incondicionalmente desde el primer momento que entre decidí embarcarse en mis estudios universitarios

A mi esposa y mi hijo quienes fueron un pilar fundamental con todo su amor y paciencia, me ayudaron a no rendirme y superarme para llegar a ser un profesional y una mejor persona

Jorge Andrés Criollo Navarrete



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AGRADECIMIENTO

A toda mi familia quienes me apoyaron y motivaron a culminar mis estudios universitarios a mi madre quien es un ejemplo de vida y fortaleza a mi padre quien me enseñó que el trabajo duro siempre tiene su experiencia y recompensa

A mi hermano quien con su apoyo y acompañamiento en todo este proceso no hubiese tenido los mejores resultados y en general a mis amigos Blanquita, Cristian Ronny y Sebastián quienes con sus ánimos, apoyo y confianza estuvieron en los momentos importantes de culminar mi trabajo de titulación

A la Universidad Técnica del Norte por abrirme sus puertas y brindarme el conocimiento necesario para poder desarrollarme como profesional y como persona, y primordialmente a mi Director Magister Luis Suarez y Asesor Magister Jaime Michilena por guiarme y ayudarme con todo el conocimiento para culminar con éxito mi trabajo de titulación

Jorge Andrés Criollo Navarrete



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
RESUMEN

El presente trabajo de titulación detalla el desarrollo de un sistema inteligente de reconocimiento de imagen a través de un algoritmo de aprendizaje de máquina para la detección de la enfermedad Erysiphe cichoracerum en cultivos Micro-Farms.

El sistema presenta el desarrollo de la parte de software como la más importante a través de la programación, así también complementado con la parte de hardware siendo estos dos bloques esenciales para poder contemplarlos y analizarlos de manera individual. En el primer bloque, el de software donde contemplamos el desarrollo en conjunto de una programación basada en lenguaje Python que en conjunto con las herramientas de entrenamiento de algoritmo de maquina basado en la estructura fundamental y principal de Tiny Yolo. Se puede desarrollar un algoritmo capaz de ser entrenado en base al reconocimiento de imágenes para que cumpla una función en específico. En el segundo bloque el de Hardware contemplamos el uso de un acelerador de GPU para correcto funcionamiento de nuestro algoritmo previamente entrenado, así como el uso de una cámara que en conjunto hacen arte de todo el sistema de reconocimiento en tiempo real de la enfermedad en los cultivos micro-farms de lechugas.

Finalmente se realizan las pruebas necesarias del sistema de reconocimiento tanto en ambientes controlados como en ambientes reales donde se contempla que todas las funcionalidades planteadas al inicio del proyecto sean resueltas con la culminación de este. Para ello se realizan pruebas a todo el sistema tanto en hardware como en software para comprobar que en la unificación de los dos no presenten errores y el sistema sea eficiente y eficaz al momento de implementarlo teniendo los resultados óptimos propuestos al inicio del proyecto.



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
ABSTRACT

This degree work details the development of an intelligent image recognition system through a machine learning algorithm for the detection of Erysiphe cichoracearum disease in Micro-Farms crops.

The system presents the development of the software part as the most important through programming, as well as complemented with the hardware part, being these two blocks essential to be able to contemplate and analyze them individually. In the first block, the software block where we contemplate the joint development of a programming based on Python language that together with the machine algorithm training tools based on the fundamental and main structure of Tiny Yolo. It is possible to develop an algorithm capable of being trained based on image recognition to fulfill a specific function. In the second block of Hardware we contemplate the use of a GPU accelerator for correct operation of our previously trained algorithm, as well as the use of a camera that together make art of the whole system of recognition in real time of the disease in micro-farms lettuce crops.

Finally, the necessary tests of the recognition system are carried out both in controlled environments and in real environments where it is contemplated that all the functionalities proposed at the beginning of the project are solved with the culmination of this one. To this end, the entire system is tested in both hardware and software to verify that the unification of the two does not present errors and that the system is efficient and effective at the time of implementation, having the optimal results proposed at the beginning of the project.

<i>Fig 1 Lechuga Red'Fire</i>	27
<i>Fig 2 Lechuga Mac</i>	27
<i>Fig 3 Lechuga Escarola</i>	28
<i>Fig 4 Lechuga Salinas o Romana</i>	28
<i>Fig 5 Lechuga Achicoria</i>	28
<i>Fig 6 Pulgón de Lechuga</i>	33
<i>Fig 7 Orugas Defoliadoras</i>	33
<i>Fig 8 Gusano Gris</i>	33
<i>Fig 9 Caracol y Babosa</i>	33
<i>Fig 10 Mosca Blanca</i>	33
<i>Fig 11 Minador de Lechuga</i>	33
<i>Fig 12 Alternaria en la Lechuga</i>	39
<i>Fig 13 Esclerotinia</i>	39
<i>Fig 14 Botrytis</i>	39
<i>Fig 15 Microdochium panattoniana</i>	39
<i>Fig 16 Erysiphe cichoracerum</i>	39
<i>Fig 17 Septoria lactucae</i>	39
<i>Fig 18 Virus del mosaico de la Lechuga</i>	41
<i>Fig 19 Sistema de Visión artificial</i>	44
<i>Fig 20 Detección de Objetos</i>	45
<i>Fig 21 Técnicas de Maching Learning</i>	47
<i>Fig 22 Aprendizaje de clasificación líneas</i>	48
<i>Fig 23 Regresión lineal</i>	48
<i>Fig 24 estructura de neuronas artificiales</i>	51
<i>Fig 25 Red Neuronal totalmente conectada</i>	52
<i>Fig 26 Estructura de una red neuronal convolucional</i>	55
<i>Fig 27 Operación de Convolución</i>	56
<i>Fig 28 Ejemplo del funcionamiento de la capa de pool</i>	58
<i>Fig 29 comparación de Yolo ante otros clasificadores</i>	60
<i>Fig 30 Modelo de Tiny Yolo</i>	62

<i>Fig 31 Fases del ciclo de la investigación</i>	<i>64</i>
<i>Fig 32 Ubicación Geográfica The Creeps</i>	<i>67</i>
<i>Fig 33 Diagrama de Bloques del sistema.....</i>	<i>83</i>
<i>Fig 34 Diagrama de Flujo del Sistema</i>	<i>85</i>
<i>Fig 35 Arquitectura del Sistema</i>	<i>86</i>
<i>Fig 36 Configuración Principal de Tiny YOLOv3</i>	<i>88</i>
<i>Fig 37 Bases de Datos Adquiridas</i>	<i>89</i>
<i>Fig 38 Base de datos en Orden</i>	<i>90</i>
<i>Fig 39 Proceso de Etiquetado</i>	<i>91</i>
<i>Fig 40 Resultado en Coordenadas del Etiquetado.....</i>	<i>91</i>
<i>Fig 41 imagen para el data set.....</i>	<i>93</i>
<i>Fig 42 Imagen de entrada</i>	<i>94</i>
<i>Fig 43 Kernel.....</i>	<i>95</i>
<i>Fig 44 Filtros para la primera capa oculta de las neuronas</i>	<i>95</i>
<i>Fig 45 Producto Matricial con la imagen de entrada del kernel</i>	<i>96</i>
<i>Fig 46 Técnicas de Max-Pooling 2x2.....</i>	<i>97</i>
<i>Fig 47 Primera Convolución.....</i>	<i>97</i>
<i>Fig 48 Mapa de Convoluciones.....</i>	<i>98</i>
<i>Fig 49 Erysiphe reconocida con el modelo entrenado.....</i>	<i>99</i>
<i>Fig 50 Cultivo Micro-Farm Diferentes Etapas de crecimiento</i>	<i>101</i>
<i>Fig 51 Cultivo Micro-Farm 3 Niveles en Vertical</i>	<i>101</i>
<i>Fig 52 Sistema de Reconocimiento De Imagen.....</i>	<i>102</i>
<i>Fig 53 Implementación del Sistema Completo con el Cultivo en Vertical.....</i>	<i>102</i>
<i>Fig 54 Arquitectura el Sistema.....</i>	<i>104</i>
<i>Fig 55 Archivos de Ejecución de Todo el Sistema</i>	<i>105</i>
<i>Fig 56 Funcionalidad en tiempo Real del Algoritmo EN base al Entrenamiento de Lechugas.....</i>	<i>106</i>
<i>Fig 57 Cuadro Delimitador de lechuga Sana o Enferma.....</i>	<i>106</i>
<i>Fig 58 Precisión el Reconociendo en Tiempo Real.....</i>	<i>107</i>
<i>Fig 59 Niveles de Cultivos Micro-Farm de Lechugas.....</i>	<i>108</i>

<i>Fig 60 Funcionamiento Cultivo Enfermo Sano</i>	<i>Fig 61 Funcionamiento del cultivo Enfermo y Sano.....</i>	<i>108</i>
<i>Fig 62 Cultivos de Lechuga de la Funca San Rafael ubicada en el sector de Imbaya</i>		<i>111</i>
<i>Fig 63 Cultivos de Lechugas etapa de Cosecha revisión de cultivos Enfermos</i>		<i>112</i>
<i>Fig 64 Plantaciones de Lechugas en Diferentes Etapas de Crecimiento</i>		<i>112</i>
<i>Fig 65 Estructura de Instalación y Creación de archivos para la Ejecución del Sistema.</i>		<i>118</i>
<i>Fig 66Rutas de los archivos para el pre-entrenamiento del algoritmo</i>		<i>119</i>
<i>Fig 67 Archivo de Ejecución Previo Activación he GPU y aceleradores Gráficos.....</i>		<i>119</i>
<i>Tabla 1 Clasificación taxonómica de la Lechuga</i>		<i>24</i>
<i>Tabla 2Contenido nutricional (unidades por 100 g de tejido de cuatro tipos de lechuga)</i>		<i>25</i>
<i>Tabla 3 Tipos de Lechugas en Ecuador</i>		<i>27</i>
<i>Tabla 4 Manejo de las Principales Plagas</i>		<i>29</i>
<i>Tabla 5 Manejo de las Principales enfermedades</i>		<i>34</i>
<i>Tabla 6 Virus de la Lechuga</i>		<i>40</i>
<i>Tabla 7 Software y Hardware de Optimización</i>		<i>42</i>
<i>Tabla 8 Algoritmos de Aprendizaje Supervisado</i>		<i>49</i>
<i>Tabla 9 Stakeholders</i>		<i>69</i>
<i>Tabla 10 Términos Abreviados y Acrónimos Usados Durante el Análisis de Requerimientos</i>		<i>71</i>
<i>Tabla 11 Priorización de los Requerimientos para el Sistema</i>		<i>71</i>
<i>Tabla 12 Requerimientos del Propietario</i>		<i>72</i>
<i>Tabla 13 Requerimientos del Sistema</i>		<i>73</i>
<i>Tabla 14 Requerimientos de la Arquitectura</i>		<i>74</i>
<i>Tabla 15 Tabla de Recursos</i>		<i>76</i>
<i>Tabla 16 Requerimientos Tecnológicos</i>		<i>76</i>
<i>Tabla 17 Selecccion de ocputador</i>		<i>78</i>
<i>Tabla 18 Selecccion de Acelerador de GPU</i>		<i>79</i>
<i>Tabla 19 Selección de Lenguaje de Programación</i>		<i>80</i>
<i>Tabla 20 Software de Tratamiento de Imagen</i>		<i>81</i>
<i>Tabla 21 Comparativas Relevantes de los Cultivos Mircro-Farm y Tradicionales</i>		<i>113</i>

INDICE

Contenido

1. CAPITULO I. EL PROBLEMA	16
1.1. Problema	16
1.2. Objetivos.....	17
1.2.1. Objetivo General.....	17
1.2.2. Objetivos Específicos	17
1.3. Alcance	18
1.4. Justificación	19
2. CAPITULO II. MARCO REFERENCIAL	21
2.1. Agricultura	21
2.1.1. Agricultura de Precisión	21
2.1.1.1. Etapas de la Agricultura de Precisión	21
2.1.2. Variedades de Hortalizas	22
2.1.3. Micro-Farms	23
2.1.3.1. Definición.....	23
2.1.4. Cultivos de Lechugas	24
2.1.4.1. Valor Nutritivo	25
2.1.4.2. Variedades de Lechuga	26
2.1.4.3. Plagas de la Lechuga	29
2.1.4.4. Enfermedades y Virus de la Lechuga.....	34
2.1.5. Equipos de Optimización y Adquisición de Datos	41
2.1.5.1. Dispositivos de Captura	41
2.1.5.2. Dispositivos de Optimización	41
2.2. Vision Artificial Intelligence	43

2.2.1. Reconocimiento de Objetos.....	44
2.2.2. Etapas para la detección de objetos en una imagen.....	45
2.3. Aprendizaje Autónomo (Maching Learning)	46
2.3.1. Técnicas de Aprendizaje.....	46
2.3.1.1. Aprendizaje Supervisado.....	47
2.4. Aprendizaje Profundo (Deep Learning).....	51
2.4.1. Arquitecturas	52
2.4.2. Redes neuronales legítimas (CNN)	54
2.4.2.1. Categorías de transformación.....	54
2.5. Lenguaje de Programación	59
2.6. Software de Entrenamiento y Procesamiento de imagen.....	59
2.6.1. Yolo.....	59
2.6.1.1. Yolo V3.....	59
2.6.1.2. Tiny Yolo	60
2.6.2. LabelImage.....	61
2.6.3. Darknet	61
3. DESARROLLO EXPERIMENTAL	62
3.1. Metodología	63
3.1.1. Action-Reserch	63
3.1.2. Ciclo de Indagación Action-Reserch.....	63
3.2. Análisis	65
3.2.1. Situación Actual	65
3.3. Introducción al desarrollo del proyecto	68
3.3.1. Propósito del Sistema	68
3.3.2. Características de los Beneficiarios.....	68

3.3.3. Ámbito del Sistema	69
3.3.4. Dimensionamiento de Stakeholders	69
3.3.5. Técnicas de Recolección de Información	69
3.4. Requerimientos	70
3.4.1. Nomenclatura de los Requerimientos	70
3.4.2. Requerimientos de Stakeholders	71
3.4.3. Requerimientos del Sistema	73
3.4.4. Requerimientos de Arquitectura	74
3.5. Recursos	75
3.5.1. Recursos económicos	76
3.5.2. Recursos tecnológicos	76
3.6. Descripción General del Sistema	76
3.7. Elección de Hardware y Software	77
3.7.1. Elección de Hardware	77
3.7.1.1. Selección de Computador	77
3.7.1.2. Selección de Acelerador de GPU	78
3.7.2. Elección de Software	79
3.7.2.1. Lenguaje de programación	80
3.7.2.2. Software de Tratamiento de Imágenes	80
3.8. Diseño del Sistema	82
3.8.1. Diagrama de Bloques del Sistema	82
3.8.1.1. Diagrama del Funcionamiento del Sistema	83
3.8.2. Diagrama de Flujo	84
3.8.3. Diagrama de la Arquitectura del sistema	85
3.8.4. Desarrollo del Software	87

3.8.4.1.	Yolo.....	87
3.8.4.2.	Configuración Darknet en Yolo Tiny.....	87
3.8.4.3.	Adquisición de imágenes	88
3.8.4.4.	Etiquetado de imágenes.....	90
3.8.4.5.	Entrenamiento del Algoritmo Mediante Redes Neuronales Convolucionales	92
3.8.4.6.	Reconocimiento de Erysiphe cichoracerum en la Lechuga	98
4.	PRUEBAS DE FUNCIONAMIENTO.....	100
4.1.	Adquisición de datos.....	100
4.2.	Pruebas del Sistema	102
4.2.1.	Diagrama de Conexión basados en la arquitectura del sistema.....	103
4.2.2.	Pruebas del sistema en tiempo real.....	105
4.2.3.	Visión por computador y aprendizaje de maquina	110
4.2.4.	Análisis de cultivos Tradicionales y Micro-Farm	111
5.	CONCLUSIONES Y RECOMENDACIONES.	115
5.1.	Conclusiones	115
5.2.	Recomendaciones	117
6.	ANEXOS.....	118
6.1.	Estructura Principal de la Programación del Sistema.....	118
6.2.	Rutas y comandos Esenciales en la Programación	119
7.	BIBLIOGRAFIA.....	120

1. CAPITULO I. EL PROBLEMA

1.1.Problema

En la actualidad la agricultura juega un papel muy importante en la economía y el desarrollo de un país, por lo que el tiempo es muy valioso. (Sowmiya et al., 2017)

De la misma manera la inseguridad alimentaria es un problema mundial que afecta a cientos de millones de personas. La Organización de las Naciones Unidas para la Agricultura y la Alimentación (FAO) estima que 821 millones de personas en todo el mundo experimentaron privación crónica de alimentos en 2017. Además, el número de personas que padecen inseguridad alimentaria ha ido en aumento desde 2014. La FAO también reconoce el efecto del cambio climático y los fenómenos meteorológicos extremos en la inseguridad alimentaria (Hatcher et al., 2019), es uno de los problemas más difíciles y urgentes de resolver dentro de una comunidad que intenta recuperarse de un desastre natural. Los PIED enfrentan un desafío de seguridad alimentaria periódico y único después de eventos traumáticos como el huracán Dorian. (Cassidy et al., 2020) (Cassidy et al., 2020)

En Ecuador las inusuales lluvias e inundaciones a causa del fenómeno “del Niño Costero” de 2016-2017 han tenido efectos devastadores, al inundar 160000 hectáreas de cultivos de los cuales 63000 se han perdido completamente y 97000 fueron afectados, estimando pérdidas de 110 millones de dólares, según cifras emitidas por el Ministerio de Agricultura ganadería, Acuacultura y pesca del Ecuador, en base a todos los destares ocurridos dentro del país se han tomado medidas de prevención para la inseguridad alimentaria.

En la actualidad en el Ecuador este tipo de cultivos han tenido mucha más acogida y aceptación en la ciudad de cuenta en la parroquia rural de El Valle al sur este de la capital de Azuay en donde se utiliza esta tecnología agrícola emergente que permite una producción altamente eficiente de vegetales verdes. Se define como "cualquier método de cultivo de plantas sin el uso de suelo como medio de enraizamiento, que implica el suministro de todos los nutrientes inorgánicos. (HCC) permite cosechar múltiples cultivos por año y permite que diferentes tipos de cultivos crezcan simultáneamente dentro del mismo ambiente controlado” (Hatcher et al., 2019), aunque también se pueden presentar enfermedades a causa de este

tipo de cultivos y la Humedad excesiva en estos una de las enfermedades presentes es la de hongos que se trasportan por esporas y por el aire, la importancia del control de hongos abarca un 85 por ciento de las enfermedades causadas por organismos dañinos. Sin embargo, algunas otras enfermedades serias provienen de organismos virales o bacterias. (Li et al., 2019) (Mojumdar & Chakraborty, 2020)

Un aporte a la solución de los inconvenientes que presenta la agricultura hidropónica o Micro-farms en el país y el mundo es la implementación de técnicas de cultivo inteligente basados en IoT en las cuales la implementación de tecnologías avanzadas de procesamiento de imagen, ayudaran con esta exploración que se crea un sistema totalmente autónomo que indudablemente puede ayudar a la Micro-farms a detectaren sus cultivos enfermedades y tener condiciones óptimas y controladas mediante la captura de imágenes mediante una cámara, extrayendo sus características, se lograra obtener un enfoque estructurado de una agricultura autosustentable, implementando una red convolucional profunda para la extracción y representación de características de imagen las cuales alimentaran la máquina de aprendizaje para el entrenamiento e interpretación posterior de la imagen para la toma de decisiones de correctivos para un óptimo crecimiento del cultivo.

1.2. Objetivos

1.2.1. Objetivo General

Diseñar un algoritmo de aprendizaje de máquina que permita tener un ambiente controlado en cuanto a las condiciones óptimas de cultivos Micro-farms mediante la toma de imágenes, e implementación de las medidas preventivas necesarias que garanticen un cultivo mucho más sano y con beneficios sustentables.

1.2.2. Objetivos Específicos

- Determinar la técnica adecuada de detección de imágenes que se ajuste al reconocimiento de cultivos y restricciones de capacidades computacionales para ser implementado en un sistema de control y monitoreo del crecimiento hortícola.
- Seleccionar un criterio de asociación de imágenes por medio de algoritmos de aprendizaje no supervisado que permitan reconocer zonas óptimas de cultivo con

respecto a datos ambientales y detectar las plagas o enfermedades que se estarían dando en dichos ambientes.

- Realizar las pruebas de rendimiento entre la visión por computador y los algoritmos de aprendizaje de máquina en la detección de condiciones óptimas de cultivo y analizar las comparativas entre productos obtenidos a través de la agricultura convencional.

1.3.Alcance

El presente trabajo de titulación tiene como objetivo diseñar e implementar un sistema embebido automatizado de monitoreo para tener un control optimo del crecimiento de cultivos en sistemas micro-farms, basado en un sistema de red neuronal y aprendizaje de máquina, con la finalidad de mejorar la producción dentro y fuera de temporada, y multiplicar el rendimiento por unidad de área en comparación con la agricultura convencional utilizando la metodología Action-Reserch

Como paso inicial al desarrollo del proyecto, se realizará una revisión bibliográfica acerca de, el ciclo de crecimiento de productos Micro-farms, cuáles son las variables que tienen mayor influencia y los valores recomendados para un desarrollo óptimo de la misma, redes de cámaras, sistemas de agricultura de ambiente controlado y las herramientas necesarias para el diseño y desarrollo del sistema. De esta base teórica se elegirá un cultivo hortícola de ciclo corto para las etapas de implementación del sistema, con el objetivo de optimizar la duración de las etapas de pruebas y entrenamiento. Además, se realizará un BenchMark con la finalidad de determinar las herramientas de hardware y software que mejor se acoplen a los requerimientos del entorno y que brinden mejores beneficios de adquisición.

Se planteará un esquema de recolección de datos de las imágenes más importantes en el cultivo Micro-farms, a través de una red de cámaras dependiendo del tamaño de cultivo y la integración de los requerimientos, los cuáles serán almacenados en una base de datos de software libre, que estará diseñada de acorde a las necesidades del sistema. La implementación del sistema deberá cubrir el control de variables indispensables eviten el crecimiento de la planta como las enfermedades, plagas que se puede suscitar, siendo esta última de importancia ya que al ser producen en un ambiente cerrado en el cual el control de

agua y luz debe ser muy exacto así mismo la detección temprana de cualquier tipo de enfermedades que le pueden ser peligrosos en los cultivos Micro-farms.

Al obtener una base de datos que, luego de ser sometida a un modelo representativo de sistemas multivariantes, pueda ofrecer un conjunto óptimo de datos para abordar los distintos posibles estados de cada una de las variables, el sistema pasaría a la etapa de entrenamiento, en donde se pretende estudiar varios algoritmos de aprendizaje de máquina y elegir el que mejor se adapte a el entorno o de ser necesario, hacer un híbrido entre los que presenten mejor desenvolvimiento sin perder de vista el coste computacional, con el fin de optimizar el uso de recursos tanto de hardware como de software y por ende el consumo energético.

De la implementación del sistema, se deberá lograr una automatización para el cultivo de ciertos productos de las Micro-farms, obteniendo productos de mejor calidad que los que ofrece la agricultura convencional, para esto se realizará un análisis comparativo de los vegetales resultantes de este último y el modelo de agricultura vertical en el que se implementará el sistema de visión por computador.

1.4. Justificación

Ecuador tiene altas tasas de crecimiento poblacional, esto conlleva a una mayor urbanización y por ende a un mayor uso de suelo (Hernández-Rodríguez et al., 2017), por lo que la oferta y demanda alimentaria en un futuro estarán en desequilibrio. En el informe nacional de la “TERCERA CONFERENCIA DE LAS NACIONES UNIDAS SOBRE LA VIVIENDA Y EL DESARROLLO URBANO SOSTENIBLE HABITAT III”, se menciona que uno de los problemas más agudos en la Sierra central, es la falta de acceso a tierra cultivable, además de que según los censos realizados por el MIDUVI para el periodo 2001-2010, muestran que al menos el 62.7% de la población está acentuada en ciudades y que la tasa de crecimiento poblacional en ciudades metrópoli, grandes y pequeñas es de 17%, 20,8% y 21% respectivamente. (Subsecretaría de hábitat y asentamientos humanos, 2015)

El control y automatización de las Micro-farms se encuentra en un avance continuo, con la finalidad de reducir los costos por mano de obra y de insumos como agua y fertilizantes, la producción de alimentos salubres, de mejor calidad y durante todo el año. Los productores

Micro-farms han observado que el desarrollo de las plantas aumenta en un 20% como consecuencia de uso inteligente de la automatización y la toma de decisiones rápidas al momento de prevenir plagas o enfermedades que se pueden dar a causas de las mismas Micro-farms y el riego que se tiene que mantener con los cultivos hortícolas, este es uno de los desafíos más importantes en entornos cerrados, ya que los riegos que se deben realizar a los cultivos tienen que ser preciso para evitar enfermedades a causa de los mismos sistemas de riego.

La práctica de Micro-farms permite solventar en cierta parte la demanda de alimentos y la reducción del suelo cultivable ocasionada por el crecimiento poblacional, ya que al hacer uso de técnicas de agricultura vertical y de precisión es posible ofrecer productos de alta calidad dentro y fuera de temporada, además de aprovechar al máximo el espacio del acre haciendo uso de las tres dimensiones y multiplicando la productividad por área de cultivo (Nwosisi & Nandwani, 2018) , esta última característica se ve apoyada por el Plan Nacional de desarrollo, en donde se menciona que, “la nueva arquitectura productiva favorecerá: El uso de tecnologías aplicadas al incremento de la productividad ...”. (Senplades, 2017)

2. CAPITULO II. MARCO REFERENCIAL

2.1.Agricultura

2.1.1. Agricultura de Precisión

La valoración de la producción agrícola en los últimos años se ha centrado en las predicciones precisas e inofensivas para el desarrollo de la producción de alimentos y satisfacer la demanda mundial adecuado a la expansión demográfica esperada en el 2050 (según la FAO). En general los parámetros ambientales que se obtienen mediante sensores son un fundamento sólido para el desarrollo actual de los cultivos impulsando actividades hacia la administración de tareas de mayor precisión en las tareas agrícolas. De la misma manera se favorecería la utilización de los beneficios del monitoreo en tiempo real.

2.1.1.1. Etapas de la Agricultura de Precisión

La agricultura de precisión permite el manejo del cultivo a través de la observación, mediciones y toma de decisiones teniendo en cuenta la variabilidad de factores que afectan los cultivos (Ruiz Gómez, 2022), así que dentro de las etapas dentro de la agricultura de precisión se presentan:

- La captación de los datos es un mecanismo tecnológico que necesita que este preinstalada en el suelo a trabajar y con la incorporación de una red de sensores se estará en la capacidad de recabar la información.
- El análisis de resultados que se realiza cuando han sido registrados los datos para luego ser interpretados matemáticamente o a base de la estadística con el mapeo, gráficos y clasificación de datos
- La toma de decisiones una vez que se tienen las principales conclusiones que fueron emitidas sobre la base de los datos recabados se tomará en consideración la decisión más apropiada la misma que la hará un ingeniero agrónomo o técnico agrícola y por último la evaluación o a su vez conocida como la etapa de monitorización del rendimiento siendo así que una vez que termina el cultivo el agricultor analizará el contexto en el cual se desarrolló la producción, por ende en

este punto se puede conocer sobre los aciertos y falencias presentadas para luego aprender de los errores.

2.1.2. Variedades de Hortalizas

En la Actualidad se encuentran una variedad extensa de cultivos de Hortalizas para el consumo humano, además estas aportan con al cuerpo vitaminas y minerales por ende son empleadas en la alimentación, de entre las más principales se pueden mencionar según Cristian (Ruiz Gómez, 2022)

- Cilantro. – el cultivo de la planta es anual, llega a crecer de 40 a 60 centímetros, sus ramificaciones son delgadas, sus flores son pequeñas, blancas o tenuemente rosadas, presentan un olor agradable y suave, pero de degustación fuerte.
- Acelga. - la siembra de las acelgas tiene la característica de ser una planta bianual de ciclo largo que no desarrolla raíz o a su vez no da un fruto comestible más bien es comestible sus hojas.
- Coliflor. – planta anual que presenta un tallo reducido y no ramifica, sus hojas son de aspecto liso, largas y con de color verde claro, la parte comestible está en los pequeños árboles que salen del tronco que son de color blanco crema, pero la tonalidad de las flores es amarilla del que se desprenden cuatro pétalos.
- Zanahoria amarilla. – el cultivo de esta planta es de ciclo anual, en el tiempo de su desarrollo se forma una roseta con escasas hojas, en la segunda estación de evolución luego del descanso se presenta un tallo corto en el mismo que se forman flores.
- Lechuga. - es una planta de cosecha anual que presenta raíz pivotante de rápido crecimiento, dependiendo del tipo sus hojas pueden ser lisas, crespas de color verde amarillento hasta llegar a una tonalidad morada.
- Rábano. – planta anual o bienal, presenta un color rojo, rosado, blanco, pardo oscuro, las hojas son basales con pequeños pelos incrustado, de borde irregularmente dentado, el fruto se presenta de 3 a 10 centímetros de longitud, de textura esponjosa, indehisciente.

- Pimiento. – el ciclo de esta planta es anual, presenta hojas enteras lampiñas y lanceoladas, las flores surgen solitarias en los nudos del tallo, el fruto es de color verde, rojo, amarillo, tomate, blanco, negro, violeta dependiendo de la variedad.
- Espinaca. –Presenta superficialmente su sistema radicular, llega a medir el producto de 80cm de largo a 30 cm de ancho, las hojas es la parte comestible las mismas que son lisas de color verde y el crecimiento es en roseta, además tiene un tallo corto.
- Cebolla colorada. – es una planta bienal, presenta un tallo reducido, el color de la piel muestra un rojo púrpura, el tamaño del producto comestible es mediano y grande la producción de estas se encuentran durante todo el año.
- Col. – la planta de este producto se presenta bianual, siendo así que su repollo es una gran yema terminal, las hojas son suaves al tacto, pero otras son crocantes, su forma es achatada, en otras ocasiones redonda o a su vez puntiaguda.

2.1.3. Micro-Farms

2.1.3.1. Definición

Una Micro-Farm son espacios de cultivo no tradicional, extensiones de tierra en donde los alimentos cultivados son los más importantes y el objetivo principal, los cuales a menudo los vamos a encontrar en áreas urbana. En este caso las micro-farms son consideradas en espacios específicos en los cuales la distribución, el diseño y la planificación en la producción de alimentos son consideradas como los enfoques principales

Las Micro-Farms pueden ser espacios productivos increíbles que resaltan el placer y los éxitos de los alimentos cultivados para una amplia audiencia. También sirve como ejemplo de que, una vez cultivado los alimentos, en lugar de agotar la tierra la mantienen para futuros cultivos, esto puede marcar una gran diferencia a la hora de mantener una planta más saludable. (*Guidelines for Growing Microgreens LESSONS LEARNED FROM ECO CITY FARMS*, n.d.)

2.1.4. Cultivos de Lechugas

La lechuga es la hortícola que más se come cruda a nivel mundial. Por sus atributos las hortalizas de hojas están expuestas a una sucesión de daños que se presentan principalmente mientras se da su etapa de postcosecha, en relación a sus procesos con fisiológicos, daños físicos y por lo tanto ataques microbiológicos, dando como consecuencias pérdidas exponenciales que pueden llegar hasta el 60% de la producción a nivel mundial.

Procedente de la India y Asia Central data de hace 2500 años una de las primeras lechugas de las cuales se tiene referencia son las de hoja suelta y las acogolladas de Europa desde el siglo XVI, en la actualidad es una de las hortalizas más cultivadas y con una gran demanda a nivel nacional e internacional.

La corta vida de postcosecha de la lechuga se da principalmente debido a su deshidratación, presentando pérdidas de turgencia y amarillamiento como consecuencia de la disminución de la clorofila. Otro de los agentes de deterioro de esta hortaliza es el pardeamiento enzimático, el cual es quien ocasiona una apariencia desagradable en el producto (Tomás-Barberdán et al., 2019)

La lechuga mínimamente procesada es una hortaliza de mayor demanda, debido al aumento en el consumo de comidas rápidas y ensaladas preparadas. Sin embargo, presenta una mayor susceptibilidad al pardeamiento enzimático. La revisión de este daño crucial para evitar las pérdidas en productos mínimamente procesados (Roura et al., 2008)

Tabla 1 Clasificación taxonómica de la Lechuga

Clasificación Taxonómica	
Reino	Plantea
División	Magnoliophyta
Orden	Asterales
Familia	Asteraceae

Subfamilia	Cichorioideae
Genero	Lactuca
Especie	Lactuca sativa L.

2.1.4.1. Valor Nutritivo

La lechuga siempre será una fuente de vitaminas y minerales; sin embargo, existe una gran diferencia en los valores nutricionales en cuanto a los tipos de lechugas (Tabla 1)

Tabla 2 Contenido nutricional (unidades por 100 g de tejido de cuatro tipos de lechuga)

Tipos	Minerales (mg)					Vitaminas		Agua (%)	Fibra (g)
	Ca	P	Fe	Na	K	A (UI)	C (mg)		
Lechuga Red-Fire	20	22	0,5	9	175	330	6	95	0,5
Romana Mac	35	26	2,0	9	264	970	8	95	05
Dancing	25	25	1.4	9	264	1.900	18	94	0,7
Escarola	25	25	1,4	9	264	1.900	18	94	0,7
Luana o Salinas	25	24	1,6	9	264	1.900	18	95	0,7
Achicoria	24	24	1,6	9	330	2.500	18	94	0,7

Fuente: Devis et al, 2002. Plagas y enfermedades de la lechuga.

2.1.4.2. Variedades de Lechuga

Montesdeoca (2009) refiere que las lechugas en la actualidad se agrupan según su forma de cultivo, lo que ha llevado a su clasificación en tres categorías: arrepolladas o arrepolladas, tienen hojas grandes alrededor de un color verde claro y son muy buscadas en el comercio. , hojas sueltas o Crispa no forma repollo, las hojas son grandes, de color verde claro, los bordes son muy rizados, se pueden recolectar en manojos individuales, no rasgue la planta y trencé o extienda, las hojas alargadas forman brotes sueltos y quebradizos, Las hojas son verde por fuera y blanco por dentro, de los cuales en Ecuador se cultivan mayormente en las provincias de Azuay, Tungurahua, Pichincha y Loja en los cuales tienen un ambiente adecuado en donde la temperatura oscila entre los 15 a 20 grados centígrados.

En los suelos ecuatorianos se cultivan seis tipos de esta legumbre las cuales son:

Tabla 3 Tipos de Lechugas en Ecuador

Tipos de Lechuga en Ecuador



Fig 1 Lechuga Red'Fire

Lechuga Red-Fire:

Sus hojas son suaves y dan un matiz especial a la preparación de alimentos. Son utilizadas en platos a la carta y tienen gran demanda en la época de Navidad. Su cabeza es blanda y grande. Se siembra sobre suelos bien desmenuzados y nivelados.



Fig 2 Lechuga Mac

Mac:

Esta hortaliza es resistente al calor. Por eso, se utiliza para las hamburguesas, porque es crujiente. Se caracteriza porque su color se mantiene verde y no se amarillenta. Su tamaño promedio es de 30 centímetros. Los naturistas la usan para hacer cremas.



Dancing:

Es una lechuga rizada y pequeña. También tiene gran demanda en el mercado y se usan para las ensaladas y decoración del plato. Además, se emplea para arreglos y se mezcla con la red fire. Es una de las lechugas cuyas hojas son las más resistentes.



Fig 3 Lechuga Escarola

Escarola:

Es de hoja rizada abierta. Su base es blanca, arrosetada, formando un bicolor verde y amarilla. Su sabor es dulce amargo, mide hasta 70 centímetros. Se consigue en los meses de invierno. Esta variedad se consume de manera especial en Europa.



Fig 4 Lechuga Salinas o Romana

Luana o Salinas:

Es el repollo más común y de mayor demanda en el mercado. Su sabor es agradable y sus hojas dan la apariencia de estar aceitosas por su color verde claro brillante. Se caracteriza porque su cabeza es firme. Se adapta a diferentes tipos de climas.



Fig 5 Lechuga Achicoria

Achicoria:

Por su sabor amargo es preferida por los extranjeros y se usa en la comida gourmet. Se recomienda hervir en agua por 10 minutos. Es alargada y de hojas finas, erectas y compactas. Los tejidos de la planta contienen abundante grasa.

2.1.4.3. Plagas de la Lechuga

Maroto (1983), cita que entre las plagas más comunes como muestra la tabla 4 estarían atacando a las lechugas y de las cuales se debería tener una consideración a la hora de realizar cultivos hortícolas Micro-Farms de Lechugas

Tabla 4 Manejo de las Principales Plagas

Plagas	Descripción	Daños	Prevención	Tratamiento
Pulgones	Se trata de una plaga sistemática en el cultivo de la lechuga. El ataque de los pulgones suele ocurrir cuando el cultivo está próximo a la recolección. Aunque si la planta es joven y el ataque es grande, puede arrasar el cultivo. También transmite virus.	mediante picaduras y succiones dañan los tejidos y produce pequeños agujeros en la hoja, puede llegar a enrollar los extremos hacia dentro. El contagio se suele producir desde las hojas exteriores hacia el interior de la planta. Esta se debilita y termina por morir.	asociación de diferentes cultivos con cebolla, ajo o tomate.	diferentes tratamientos, como las piretrinas, el uso de jabón potásico o los enemigos naturales como las mariquitas o repelentes naturales.
Orugas defoliadoras	Proceden tanto de las polillas nocturnas como de las mariposas diurnas. Estos insectos ponen sus huevos en la planta de la que se alimentan, y más tarde, de los huevos surgen	pequeños agujeros en las hojas, causando defoliación. Comienza destruyendo el envés de la hoja, donde se suele esconder, pero al crecer, sus mandíbulas		Tratamiento: aplicación de insecticidas biológicos cada semana. Se puede usar <i>Bacillus thuriengensis</i>

	las orugas, que también se alimentan de la planta sin parar hasta que forman la pupa. Las poblaciones tienden a ser máximas en otoño. Pueden destruir la yema apical de crecimiento, deteniendo el desarrollo de la planta.	(bacteria cuya toxina únicamente daña el tracto digestivo de estas orugas, aunque como no afecta a los huevos habría que repetir la aplicación varias veces), Deltametrina y otros productos combinados que se están testando actualmente.
Gusano Gris	Es un gusano de gran tamaño que suele encontrarse en zonas frescas y húmedas, se alimenta de noche y durante el día permanece enroscada enterrada en el suelo. Son más comunes en primavera.	atacan la base de los tallos y las raíces de las plantas, llegando a secarlas y provocando la caída de la plántula. entre sus enemigos naturales destaca la familia Carabidae, una familia de coleópteros. También se pueden poner trampas o hacer tratamiento con agentes químicos (tratamiento fitosanitario). Se

				suele usar Deltametrin, lambda cihalotrin o azadiractin. Estos tratamientos se llevarán a cabo en lechuga cuando más del 2% del cultivo esté dañado.
Caracoles y babosas	Los caracoles y babosas son Gasterópodos que viven en lugares húmedos. Por lo general, no suelen estar presentes durante el invierno, suelen aparecer en primavera y otoño.	he alimentan de las hojas de la lechuga	hay plantas que repelen a las babosas y los caracoles, como el tomillo, el eucalipto, la menta, la lavanda o el espliego. Podemos plantarlas alrededor de nuestros cultivos para evitar el ataque de estos animales.	se puede usar fosfato férrico, que no es tóxico, que las plagas se comen y produce la falta de apetito de caracoles y babosas, de modo que dejan de alimentarse hasta que mueren.
Minadores	Si has observado galerías en las hojas de lechuga,	retrasa el inicio de la maduración de las	L a infestación se puede	son difíciles de controlar dado

seguramente causadas por:	serán	plantas. Excava galerías en las hojas al tiempo que se alimenta del parénquima, lo cual debilita considerablemente a la planta y provoca el marchitamiento y la caída de las hojas. Además, los daños que producen en las hojas pueden suponer una posible entrada a otro tipo de enfermedades en la planta.	prevenir mediante cultivos trampa, cultivos que atraen a estos insectos y los mantienen alejados de nuestro cultivo de interés. Podríamos usar en este caso plantas de cenizo o de aquilegia. Además, podemos poner trampas con feromonas.	que son capaces de protegerse de los insecticidas y de las defensas propias de la planta escondiéndose dentro de las propias hojas.
Mosca Blanca	Produce un debilitamiento general de la planta picando y absorbiendo los jugos fotosintéticos.	aparecen manchas oscuras y redondeadas con círculos concéntricos en las hojas. Estas manchas necróticas aparecen primero en las hojas inferiores de la planta y se extiende a partir de	se reduce el nivel de incidencia de la infección al evitar madurez temprana, evitando humedad	se pueden usar fungicidas con Cimoxanilo, o Mancozeb o Zineb. Deben repetirse las aplicaciones cada 10 o 15 días.

ahí. Puede causar la excesiva del
caída de las hojas y la follaje.
muerte precoz

Nota Fuente: (Lactuca et al., 2013), (*Botanica - Agricultura - Libro - Horticultura Herbacea Especial (Maroto Borrego JV - Mundi Prensa 1983) PDF | PDF, n.d.)*)



Fig 6 Pulgón de Lechuga



Fig 7 Orugas Defoliadoras



Fig 8 Gusano Gris



Fig 9 Caracol y Babosa



Fig 10 Mosca Blanca



Fig 11 Minador de Lechuga

Fuente: (*Plagas y Enfermedades Más Importantes de La Lechuga - NEVAL. Ensayos de Eficacia, n.d.)*)

2.1.4.4. Enfermedades y Virus de la Lechuga

Maroto (1983), cita que entre las enfermedades que se pueden dar en los cultivos de lechuga las más comunes causadas por hongos y virus las encontramos en la tabla 5 a continuación

Tabla 5 Manejo de las Principales enfermedades

Enfermedad	Descripción	Daños	Prevención	Tratamiento ²
Alternaria	Enfermedad causada por el hongo <i>Alternaria dauci</i> . Suele desarrollarse en condiciones de alta humedad, con temperaturas entre 25-28°C y alternancias entre sol y lluvia, por lo que suele haber más en primavera.	aparecen manchas oscuras y redondeadas con círculos concéntricos en las hojas. Estas manchas necróticas aparecen primero en las hojas inferiores de la planta y se extiende a partir de ahí. Puede causar la caída de las hojas y la muerte precoz	se reduce el nivel de incidencia de la infección al evitar madurez temprana, evitando humedad excesiva del follaje.	se pueden usar fungicidas con Cymoxanilo, Mancozeb o Zineb. Deben repetirse las aplicaciones cada 10 o 15 días.
Esclerotinia	Sus síntomas son como una vellosidad blanca en la base de la lechuga y como unas costras negras. Es un hongo ascomiceto que produce muchas	afecta fundamentalmente en el cuello de la planta y se va extendiendo por el resto de la planta, que termina por pudrirse. Aparecerá una masa blanca algodonosa y unos corpúsculos	se pueden hacer aplicaciones de fungicidas preventivos para prevenir el ataque por ascosporas.	se usan materias activas como el metalaxil o tiram. También se pueden usar métodos, como la solarización previa a la

	enfermedades en negros en las hojas de cultivos. lechuga	plantación y por supuesto en verano.
	Produce la enfermedad conocida como podredumbre blanca. El desarrollo óptimo de este fitopatógeno ronda los 20°C en condiciones de humedad elevada.	
Botrytis	Por todos conocido este hongo, ya que es el más común junto a <i>penicillium</i> en alimentación. Este hongo tiene un aspecto amarronado a grisáceo con vellosoidad. Hongo ascomiceto patógeno también conocido como podredumbre gris. Es más común en climas suaves con una humedad ambiental elevada. Durante el invierno, el hongo hiberna en el suelo en forma de nóduo.	produce la necrosis y el reblandecimiento de aquellos tejidos a los que ha infectado. La infección se produce a través de alguna herida en la planta, causada por insectos conviene una buena ventilación de los cultivos para evitar que la humedad quede entre las hojas. Tan pronto como aparezca alguna planta infectada deberá ser retirada porque la enfermedad se expande muy rápidamente. se pueden usar enemigos naturales. Hay muchos hongos y bacterias que atacan al patógeno. Podemos destacar <i>Trichoderma harzianum</i>, que además no tienen ningún efecto negativo sobre animales u otras plantas. El modo de acción de este hongo supone un aumento la capacidad de la

planta para absorber los nutrientes necesarios para luchar contra el patógeno.

También se puede tratar con agentes químicos fungicidas como Iprodiona o Cimoxanilo.

Microdochium panattoniana o Antracnosis en lechuga	Es un hongo fitopatógeno que produce una enfermedad conocida como antracnosis. Suele aparecer sobre las hojas más viejas de la planta, mucho antes que en el resto. Suele alojarse en el nervio central, en el peciolo y en el limbo. Suele actuar en primavera y en verano.	sobre las hojas aparecen manchas amarillentas con el margen rojizo o necrótico. El margen rojizo se va expandiendo hacia el interior, necrosando la mancha, que se convierte en un agujero. No es una enfermedad grave, pero supone un problema estético de cara a la comercialización.	la principal forma de prevención es la limpieza de los cultivos, arrancando las malezas, hojas muertas y desinfectar el suelo para que la infección no se pueda extender.	se pueden aplicar agentes químicos antifúngicos, siendo los más comunes aquellos que tienen un componente derivado del cobre.
--	--	---	---	---

Erysiphe cichoracerum u Oidio en Lechuga	Hongo ectoparasítico causante de la enfermedad Oídio, muy extendida por gran variedad de cultivos.	se desarrolla tanto en el haz como en el envés de la hoja, así como en los tallos. La cubre de un micelio de color blanco de aspecto pulverulento. Las hojas se tornan amarillas y luego se secan. No suelen causar la muerte de la planta, pero la debilitan y necrosan. Esto se debe a que los micelios bloquean sus estomas y dificultan la respiración.	existen variedades de especies resistentes al oídio. Podemos controlar los niveles de nitrógeno en suelo porque favorecen la aparición de la plaga.	podemos controlar esta enfermedad con aplicaciones de polvos minerales de azufre o cobre, o con preparados de origen vegetal como la cola de caballo (Equisetum Hyemale) o purín de Salvia.
	Por contra de lo que se pueda pensar, su presencia es más común con niveles de humedad altos y excesivo consumo de nutrientes.			También podríamos añadir micorrizas a nuestros cultivos, aunque esto es incompatible con la aplicación de fungicidas. El uso de micorrizas contribuye a la

absorción de
nutrientes por la
planta,
contrarrestando
los efectos de la
plaga.

Septoria lactucae Septoriosis en lechuga	Es un hongo patógeno o de las hojas de las lechugas. Requiere de épocas de humedad.	suele afectar primero a las hojas más viejas de la planta. Produce manchas pequeñas e irregulares sobre la parte inferior de las hojas de la lechuga. Conforme se desarrolla el hongo, estás manchas pueden llegar a ser necróticas.	los principales métodos de prevención de la enfermedad son el control de la humedad en el cultivo y la limpieza del mismo, de modo que no haya malas hierbas y follaje.	podemos usar agentes químicos fungicidas. Podemos destacar Cimoxanilo, productos cuprosos y Mancozeb.
---	--	--	---	--

Nota Fuente: (Botanica - Agricultura - Libro - Horticultura Herbacea Especial (Maroto Borrego JV - Mundi Prensa 1983) PDF | PDF, n.d.)



Fig 12 Alternaria en la Lechuga



Fig 13 Esclerotinia



Fig 14 Botrytis



Fig 15 Microdochium panattoniana



Fig 16 Erysiphe cichoracearum



Fig 17 Septoria lactucae

Fuente: (*Plagas y Enfermedades Más Importantes de La Lechuga - NEVAL. Ensayos de Eficacia, n.d.*)

Ademas, los de lechugas no solamente pueden presentar enfermedades del tipo viral que son mas dificiles de tratar, de las cuales entre la mas importante destaca la reflejada en la tabla 5.

Tabla 6 Virus de la Lechuga

Enfermedad	Descripción	Daños	Prevención	Tratamiento
Virus del mosaico de la Lechuga	Virus que más afecta a plantas en cualquier etapa de desarrollo. Se transmiten por jóvenes aparecen clareos de las nerviaciones en las hojas, así como deformación del limbo y algunos puntos necróticos. Se transmiten por vectores (suelen ser insectos o nematodos).	En las plantas más maduras aparecen deformaciones de la hoja, y manchas amarillentas. Surgen también puntos de necrosis.	Dado que no es una enfermedad tratable, es muy importante la prevención. Es importante la limpieza y desinfección del material de podar para asegurarnos que está libre del virus, así como controlar las plagas de insectos que puedan ser transmisores de estos virus. Se deben eliminar las plantas enfermas de inmediato para que no se extienda.	Dado que existen variedades de lechuga resistentes al virus del mosaico, sería recomendable escoger estas a la hora de plantar en lugar de las que no lo son.

Nota Fuente: (Botanica - Agricultura - Libro - Horticultura Herbacea Especial (Maroto Borrego JV - Mundi Prensa 1983) PDF | PDF, n.d.)



Fig 18 Virus del mosaico de la Lechuga

2.1.5. Equipos de Optimización y Adquisición de Datos

Para lograr los objetivos planteados es necesario estar al tanto de algunas de las herramientas que nos permitirán las interacciones que se tendrán entre el usuario y el sistema, teniendo en cuenta esto se detallan las herramientas idóneas que nos permitirán la captura de las imágenes y a su vez poder transmitir las, por lo cual vamos a presentar cada una de estas.

2.1.5.1. Dispositivos de Captura

En la actualidad, se puede revisar que existen muchas herramientas que son de gran ayuda al momento de la obtención de información de fotografías, teniendo en cuenta este contexto, el sistema de visión artificial necesita una gran cámara funcional para abstraer en diferentes campos diferentes aplicaciones. Como resultado la cámara a utilizar se la implementara con la finalidad de inspeccionar, monitorizar, agricultura de precisión e ingeniería, a continuación, en la tabla 7 muestra algunas cámaras a utilizar para el desarrollo del presente proyecto.

2.1.5.2. Dispositivos de Optimización

En el presente, podemos mencionar que existen una variedad de optimizadores a la hora de desarrollar un sistema de Visión Artificial (AI), por lo cual se mencionan dos de los más ocupados en la actualidad para obtener un mejor rendimiento de procesamiento de datos y así tener un mejor resultado en cuanto al tiempos de respuesta del sistema en funcionamiento.

Tabla 7 Software y Hardware de Optimización

Software o Hardware	Descripción	Especificaciones
The Intel Movidius 2	Lleva a la visión por computador y la inteligencia artificial a todos los dispositivos de IoT. Intel® Distribution of OpenVINO™ toolkit Intel Distribution of OpenVINO™ toolkit es el kit de desarrollo de software predeterminado para optimizar el rendimiento, integrar la inferencia de aprendizaje profundo y ejecutar redes neuronales profundas (DNN) en las unidades de procesamiento de visión (VPU) de Intel Movidius.	- Vision Processing Unit (VPU) The Intel Movidius - Operating Systems Support Ubuntu , Windows, Centos - Conectividad USB 3.1 Tipo A - Temperature de Operación 0 -40 C - Distribución OpenVino
CUDA y CUDNN	CUDA es un modelo de programación paralela escalable y un entorno de software para computación paralela. Extensiones mínimas al entorno familiar C/C++, modelo heterogéneo de programación en serie-paralelo La arquitectura TESLA de NVIDIA acelera CUDA y expone la potencia computacional de las GPU NVIDIA CUDA también se asigna bien a CPU multinúcleo	- Modelo de programación CUDA - Fundamentos de la programación CUDA - Pila de software - Gestión de datos - Ejecutando código en la GPU - Bibliotecas CUDA - BLAS - FFT

Nota Fuente: (Caffe, 2019),(Ruetsch & Oster, 2008)

2.2.Vision Artificial Intelligence

Dentro de las aplicaciones más importantes de la Inteligencia Artificial se han ido desatollando estas técnicas las cuales se adaptan a las necesidades del ser humano, donde (José Núñez Sánchez-Agustino et al., 2016) la define como: “una rama de la inteligencia artificial (AI) el cual su propósito es diseñar sistemas informáticos capaces de entender e interpretar características de una imagen del mundo real” (p.9), al tener en cuenta este principio este tipo de sistemas se integran en procesos que permiten extraer información tanto numérica como simbólica a partir de solo una imagen.

Acerca de estos procesos se puede identificar los diferentes campos fundamentales como se muestra a continuación en la figura en donde la visión artificial está estrechamente relacionada con las técnicas de procesamiento de imágenes y de reconocimientos de patrones

Adquisición de imágenes, mediante una cámara fotográfica u otro tipo de sensor.

Identificación y clasificación de objetos y estructuras en las regiones de interés.

Tratamiento de las imágenes para destacar posibles regiones de interés y eliminar información no relevante.

Aislamiento de las regiones de interés para su posterior análisis e interpretación.



Fig 19 Sistema de Visión artificial

Fuente: (José Núñez Sánchez-Agustino et al., 2016)

2.2.1. Reconocimiento de Objetos

Para un sistema de Inteligencia Artificial la etapa del reconocimiento de objetos es una de las más importantes al tener muchas aplicaciones, relacionada a la detección de multitudes (Abril Cañas et al., 2007), Principalmente es una de las técnicas que se usa para vincularla el momento de facilitar y detectar áreas de interés en las imágenes al considerar estos patrones permitiendo así que se realicen procesos que tiene otras características ayudando a la optimización de todo el sistema (José Núñez Sánchez-Agustino et al., 2016)

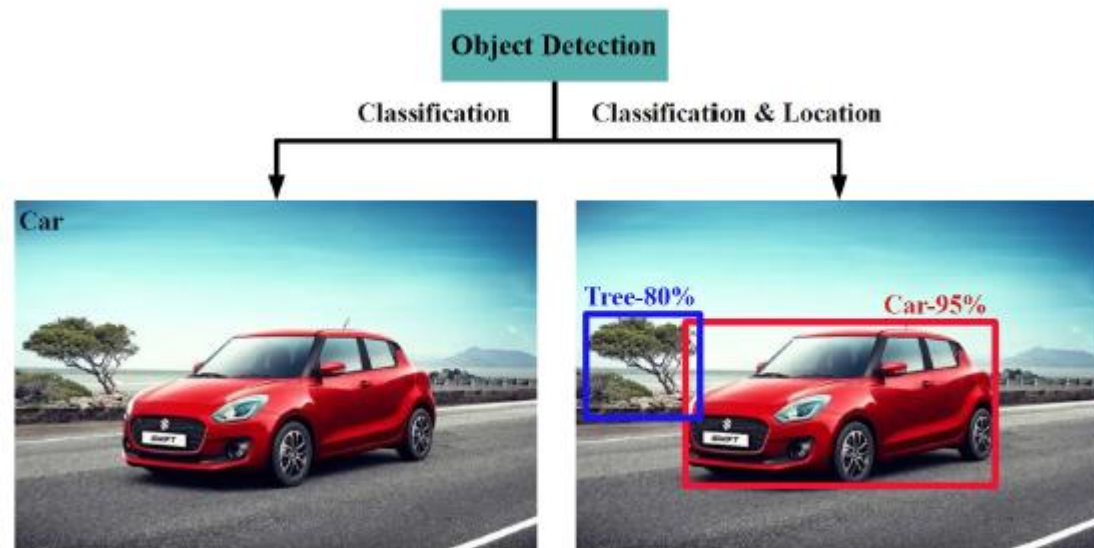


Fig 20 Detección de Objetos

Fuente: (Li et al., 2020)

En relación de la identificación de objetos se debe considerar si el sistema funciona al distinguir la presencia del perro en la etapa de clasificación de las imágenes y luego en la etapa de detección para saber cuál es el perro de acuerdo con las características propias tanto de morfología o comportamiento, como se muestra en la figura 21, por lo tanto, el sistema reconoce de acuerdo con la imagen ingresada.

2.2.2. Etapas para la detección de objetos en una imagen

- **Adquisición de la imagen:** Para obtener una imagen con un rendimiento óptimo, el sistema debe estar integrado: una cámara, un escáner, una luz. Es importante señalar que esta etapa ayuda a determinar principalmente factores como la posición de la cámara, el ruido fotoeléctrico externo (Robots, 2009) y la iluminación de la escena. Es importante corregir cualquier problema antes de implementar un algoritmo complejo. (Antontio & C., 2006; Guerrero, 2015)
- **Procesamiento de la Imagen:** En esta etapa se realiza el procesamiento de la imagen porque sufre efectos de degradación de bajo nivel y aparece como ruido y pérdida de nitidez y resolución (Robots, 2009) sirve para identificar algunas regiones o regiones de interés y descartar información que ya no se requiere (Antontio & C., 2006), a la

hora de tomar muestras se deben considerar varios factores como la mejora del contraste, la reducción de ruido, la estandarización, la iluminación y la conversión del espacio de color, entre otros (Pineda, 2018)

- Segmentación: implica etiquetar una imagen que tiene una limitación importante porque se deben tener en cuenta ciertas condiciones a la hora de crear una tarea compleja (José Núñez Sánchez-Agustino et al., 2016), implica aislar las áreas que te interesa analizar mejor en el proceso e interpretación, una imagen binaria se genera, donde se hace referencia a Los píxeles pertenecen al objeto con uno, mientras que los píxeles no pertenecen al objeto definido por cero. (Antontio & C., 2006), En otras palabras, se considera que estos píxeles tienen partes de significado idénticas a propiedades como el color (Núñez, n.d.)el movimiento (Robots, 2009)y la forma entre otros puntos. Reconocimiento de objetos: Una vez seleccionada esta técnica de clasificación, en el proceso se identifican los objetos en la entrada, la imagen pasa por el clasificador y se pueden reconocer los objetos. Y los resultados obtenidos al mostrar el tema de interés con mayor precisión. Finalmente, el clasificador define un objeto como un insumo que forma parte de la actividad del proceso (Pineda, 2018).

2.3. Aprendizaje Autónomo (Maching Learning)

Es una rama de la inteligencia artificial cuyo objetivo es hacer que las máquinas funcionen en paralelo con el software, tal como lo definen (Mohammed et al., 2016) como: “El aprendizaje automático es un corolario de la intersección de la informática y la estadística” (p. 5), por lo que es útil poder explicar algunas tareas humanas por medio de algoritmos que conduzcan a llenar este vacío de presión.

2.3.1. Técnicas de Aprendizaje

Bajo el aprendizaje automático, se identifican ciertas tecnologías que pueden aprender de manera realista ciertos modelos para lograr un objetivo. La figura 22 ilustra las técnicas de aprendizaje automático.

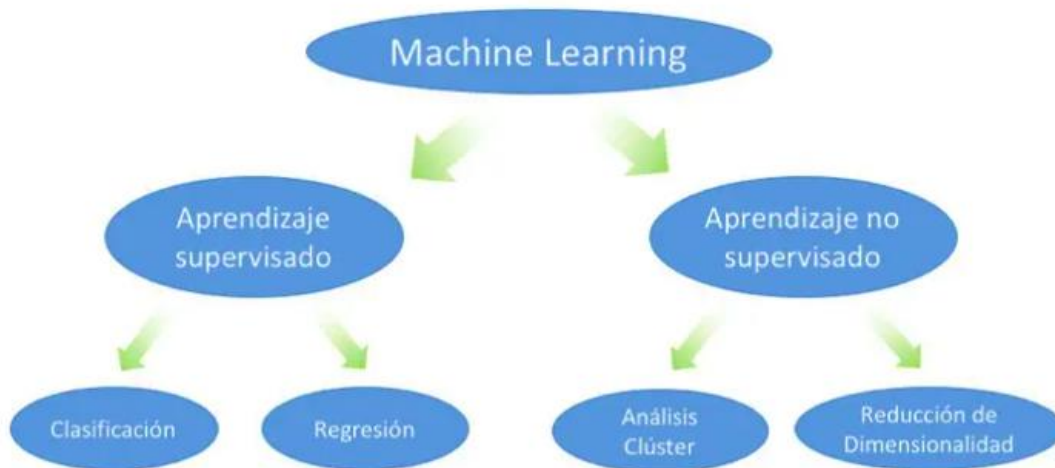


Fig 21 Técnicas de Maching Learning

Fuente: (Matich Dami, 2011)

2.3.1.1. Aprendizaje Supervisado

La funcionalidad del aprendizaje supervisado resuelve en inferir desde los datos de entrada y salida, previamente clasificados, luego se entrena al algoritmo (M. & Montilla, 2019) mediante la modificación de los pesos de la sinapsis para tener un resultado óptimo (Pineda, 2018)

En relación con su definición se encuentra una clasificación en relación con el dominio de su salida, y se obtiene dos tipos de algoritmos como son:

- **Clasificación:** En relación con el resultado de sus muestras es de manera discreta. Es decir, una serie de eventos que pueden ser ciertos, como se muestra en la figura 21, como ejemplo: Predecir si estará despejado, nuboso o lloverá (M. & Montilla, 2019)

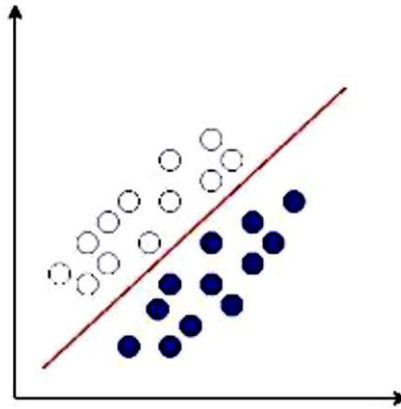


Fig 22 Aprendizaje de clasificación líneas

Fuente: (M. & Montilla, 2019)

- Regresión: Alternativamente, se requiere este método si se trata de un valor continuo. Cualquier evento ininterrumpido, por ejemplo, en la Figura 24: Venta del valor del activo.

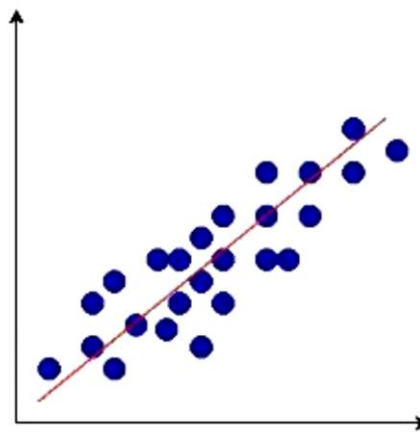


Fig 23 Regresion lineal

Fuente: (M. & Montilla, 2019)

Los algoritmos utilizados para aprendizaje supervisado se muestran en la tabla 9.

Tabla 8 Algoritmos de Aprendizaje Supervisado

Algoritmo	Descripcion	Ventajas	Limitaciones
Bayesianos	Su funcionamiento está basado en el teorema de Bayes que está relacionado a la probabilidad se genera un modelo predictivo. Los más utilizados son: Naive Bayes, Gaussian Naive Bayes, Multinomial Naive Bayes, Bayesian Network (M. & Montilla, 2019)	Óptimo para predecir clases de datasets de prueba, funciona bien en predicciones multiclase.	Necesita una gran cantidad de datos para hacer estimaciones confiables de la probabilidad de cada clase
Redes Neuronales	Es una colección de nodos llamados neuronas artificiales, los datos de entrada que recibe luego los tiene que procesar y enviar a otra neurona, al formar un nodo que ajusta el peso. (Carlos & Pérez, 2018)	Las redes neuronales pueden ser añadidas diferentes capas para realizar diferentes transformaciones de sus entradas. (Pineda, 2018)	Dependen mucho del hardware, ya que, según la arquitectura de la red neuronal, y el tamaño del conjunto de datos que se vaya a analizar se necesita mayor procesamiento para que el entrenamiento y detección no sean lentos
Árbol de decisión	Su estructura es en forma de árbol, la cual permite realizar predicciones mediante un conjunto de instancias de entrenamiento a su vez se genera un modelo	Simplifica el número de variables independientes y asigna valores específicos, a problemas, decisiones, y resultados de cada	Son sensibles al ruido lo cual podría afectar al más mínimo cambio de los datos en el árbol, podría modificarlo completamente, así como

	predictivo. (Carlos & Pérez, 2018)	decisión lo cual también las decisiones disminuye (Morales & Escalante, 2012) ambigüedades en la toma de estas. (Morales & Escalante, 2012)
Máquina de vectores de soporte (SVM)	Usados para aprender sobre la clasificación de dos clases diferentes de puntos de entrada. Su funcionamiento radica en obtener información de los vectores de soporte para formar una frontera de decisión.(BETANCOURT , 2005)	El entrenamiento es relativamente fácil. No hay óptimo local, como en las redes neuronales. El compromiso entre la complejidad del clasificador y del error pueden ser controlado explícitamente. (BETANCOURT, 2005)
KNN	Su función reside en la vecindad, al hacer referencia al número de vecinos, es decir, engloba la simplicidad de categorías (De-La-Hoz et al., 2019)	K-NN realiza las predicciones en base a la en los ejemplos más parecidos al que hay que predecir. Es muy tolerante al ruido. (Serrano et al., 2016)

Nota. Fuente: (Carlos & Pérez, 2018)

2.4. Aprendizaje Profundo (Deep Learning)

En el campo del aprendizaje automático, el aprendizaje profundo se integra porque implica una serie de limitaciones debido a la imposibilidad de procesar los datos en su forma original, lo que permite que los modelos informáticos compuestos por muchas capas aprendan datos con múltiples niveles de abstracción. (LeCun et al., 2015), lo que significa que se necesita mucha intervención humana para determinar qué patrones debe detectar la máquina en la imagen. Esto se resuelve mediante el aprendizaje profundo o deep learning al identificar automáticamente las características más destacadas y ajustar la información en bruto para que el clasificador pueda generar la salida del sistema (Aguirre, 2019)

Para entender este tema, debes conocer el concepto de red neuronal artificial, un modelo informático cuya unidad es una neurona artificial, muy similar a las neuronas de un cerebro biológico. Las secciones básicas, que se muestran en la Figura 25, incluyen la entrada, el peso y el núcleo de la neurona que incluirá la función de activación y su valor de salida (E & Briega, 2012)

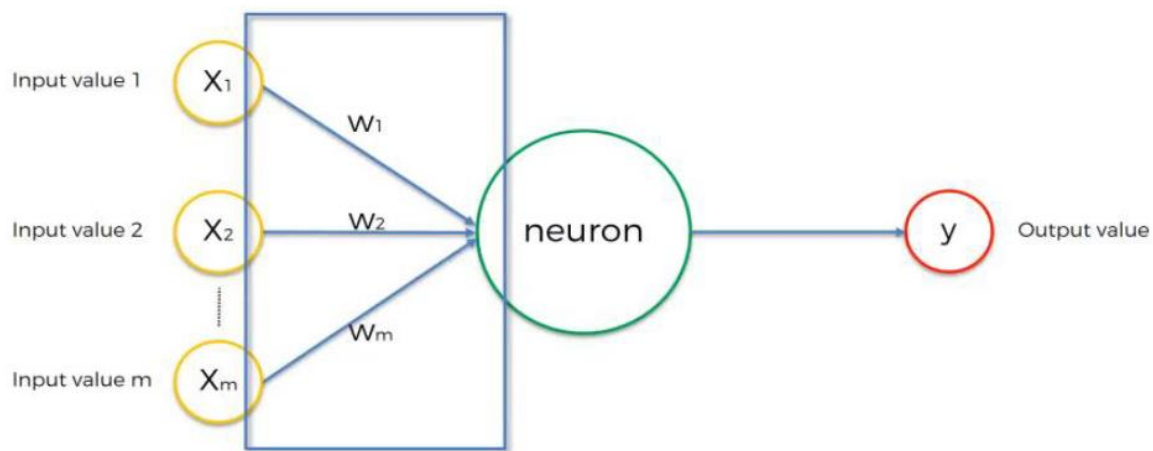


Fig 24 estructura de neuronas artificiales

Fuente: (Muñoz & Blanco, 2018)

La actividad de la neurona es la suma de los pesos de los valores introducidos en ella, compara las cantidades con el valor umbral de prueba, y si es igual o superior a este valor,

envía un mensaje de activación a la parte superior. Los pesos de los valores de entrada dependen de los pesos asignados a las conexiones entre neuronas, que determinan la fuerza con la que cada variable de entrada afecta a las neuronas (Patterson & Gibson, 2017)

Ahora bien, esta unidad, interconectada con otras neuronas, forma una red que consta de tres capas básicas (esto puede variar de un caso a otro) como se muestra en la Figura 24. Capa de entrada, datos de recepción global o variables de entorno externas, esta está conectada a la capa La capa oculta que no tiene conexión con el exterior y sus neuronas pueden estar conectadas de diferentes formas dependiendo de la topología de la red neuronal, y esta capa es la encargada de procesar y extraer las propiedades de los datos obtenidos de la capa anterior (Matich Dami, 2011), A su vez, la capa oculta conectada a la capa de salida determina el valor que se pasa a las neuronas asociadas (Borovkov, 2017)

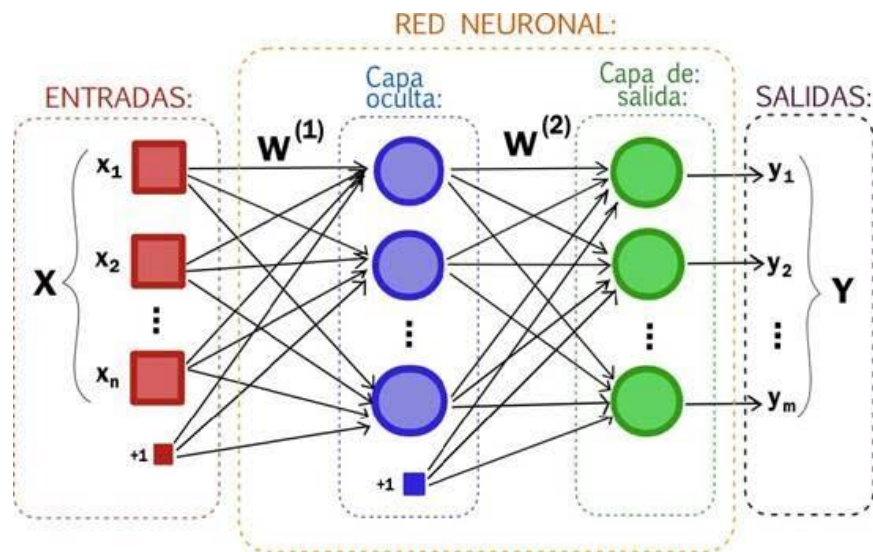


Fig 25 Red Neuronal totalmente conectada.

Fuente:(Matich Dami, 2011)

2.4.1. Arquitecturas

Dado que existen algunos escenarios donde no se conoce la estructura y el algoritmo para cada caso, se han estudiado algunos constructos y algoritmos que permitan desarrollar el proceso de entrenamiento de manera eficiente y eficaz para obtener los mejores resultados.

Además, utilizando el algoritmo más realista, se combinan las arquitecturas más populares para crear modelos de aprendizaje profundo, que se mencionan a continuación:

- Redes neuronales convolucionales (CNN)

Sus arquitecturas son muy populares debido a la ventaja de la velocidad durante la formación de la red, por lo que ayuda en la formación de redes profundas con muchas capas, y por esta razón es uno de los primeros modelos de aprendizaje profundo, y se utiliza a menudo para reconocer objetos en imágenes.

Debemos agregar que CNN utiliza tres ideas básicas: dominios locales receptivos, pesos compartidos y agregación (Neapolitan, 2018). Por lo tanto, pueden reconocer características automáticamente a partir de los datos de entrada, lo que elimina la extracción manual de características. Su estructura se basa en muestrear una parte de la imagen, descubrir sus características y construir una representación a partir de ella (Goyal et al., 2018).

- Redes neuronales recurrentes (RNN)

El funcionamiento de estas redes depende de la retroalimentación entre todos los elementos que la componen. Una característica clave es que utiliza un retraso en su salida, lo que permite una fácil predicción. Además de la dificultad de este tipo de red, también es algo complicada, ya que su funcionamiento requiere de un algoritmo básico y puede transmitir datos a lo largo del tiempo, generando así eventos, especialmente utilizados en aplicaciones como el reconocimiento secuencial de patrones. por sus beneficios (Oropeza, 2007) Las RNN resuelven este problema porque pueden procesar secuencias independientemente de su tamaño y determinar la asociación entre elementos de esa secuencia (Liu et al., 2016). El concepto de RNN es que permite almacenar y compartir información entre versiones de tiempo específicas (Patel & Patel, 2018)

- Redes de creencias profundas (DBN)

Este tipo de red consta de varias capas. Para saber a qué capa pertenece la neurona, se debe determinar el origen de la señal de datos de entrada y hacia dónde se dirige, es decir, su salida, es decir, recibir la señal a través de la entrada de otras neuronas llamadas conexiones

directas (Matich Dami, 2011). El entrenamiento de DBN se puede dividir en dos fases: entrenamiento previo no supervisado con muestras no etiquetadas y luego modificación supervisada con muestras etiquetadas (Patel & Patel, 2018). DBN se utiliza en el campo del reconocimiento de imágenes, la recuperación de imágenes, la comprensión del lenguaje natural y la predicción de defectos (Jones, 2017)

2.4.2. Redes neuronales legítimas (CNN)

Hoy en día, están ganando popularidad principalmente en el campo del reconocimiento de patrones en imágenes. Tienen una excelente capacidad para resolver problemas de clasificación de imágenes, que se aprenden de una gran cantidad de bases de datos (Wu, 2017). CNN es una red jerárquica de múltiples capas, que permite la extracción de características en capas ocultas. Constan de 5 a 7 capas, que son muy similares en la arquitectura de las redes neuronales tradicionales como cognición multicapa que combina pesos y sesgos para el aprendizaje. (E & Briega, 2012)

La CNN toma una entrada desde la cual las características se pueden extraer de forma incremental de un mapa de características de alta resolución, de modo que estas características se puedan combinar con un mapa de características de alta resolución. Baja resolución, en el proceso se extiende a capas más profundas y es adecuado para aplicaciones donde los datos están en forma de matriz, por ejemplo, imágenes que evitan procedimientos adicionales (Cong & Xiao, 2014). Las CNN procesan las imágenes por partes, lo que reduce la cantidad de parámetros requeridos en comparación con el procesamiento clásico de redes neuronales, por lo que se utilizan en aplicaciones de visión artificial (Vizcaya, 2018).

2.4.2.1. Categorías de transformación

CNN es una arquitectura de entrenamiento de varias etapas. En la arquitectura de estas redes existen dos tipos de capas, la primera cumple la función de extracción de características y la segunda se integra en el proceso de clasificación (Garc a, 2018). Las entradas y salidas de cada paso son conjuntos de matrices llamadas mapas de características, que representan una característica particular extraída en todas las ubicaciones de entrada (imagen) (Lecun et al., 2010). Cada capa de transformación de bloque de entrada con una serie de filtros se define

como un parámetro de hiperconvolución. La ejecución de estos filtros funciona en el momento en que se atraviesa la imagen y se obtiene un resultado con un valor específico, que reduce el tamaño de la imagen real llamada mapa de objetos por la cantidad de filtros aplicados (García, 2018). Los hiperparámetros que componen el proceso de convolución son:

- Número de filtro (k): Se define para una característica en particular, por lo que se aplican varios filtros para lograr un objetivo.
- Tamaño del filtro (F): Depende de los parámetros del revelador, debido a las propiedades del filtro y la velocidad de procesamiento.
- Stride (S): se utiliza para el tamaño del desplazamiento o trayectorias de los filtros de las capas por parte de los píxeles integrados en la imagen.

La estructura de una red neuronal acumulativa se muestra en la Figura 27:

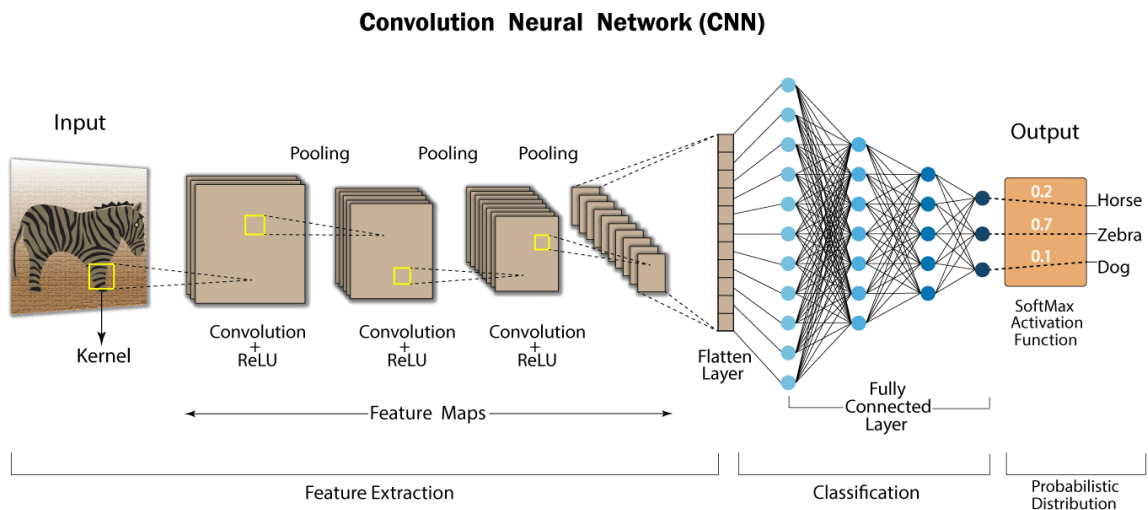


Fig 26 Estructura de una red neuronal convolucional.

Fuente: (E & Briega, 2012)

2.4.2.1.1. Capa de Entrada

Las capas de entrada son donde los datos de la imagen se recopilan y almacenan en su estado original para su procesamiento en la red. Esta entrada especifica el ancho, alto y número de canales, generalmente tres, de los valores RGB para cada píxel (Patterson & Gibson, 2017)

2.4.2.1.2. Capa Convolutiva

El propósito de esta capa es extraer características de una imagen. Para ello, multiplica elemento por elemento (producto punto) entre el submatriz de la matriz de entrada y el filtro (también conocido como campo de recepción o kernel) con un tamaño igual al tamaño del subarreglo y siempre menor que el de la matriz de entrada. Luego se suman los valores obtenidos y el resultado es un mapa de los parámetros de salida (Cha et al., 2017). El kernel filtra los datos de entrada para ciertos tipos de información y permite extraer la misma característica en cualquier parte de los datos de entrada (Acuña, 2019); Por ejemplo, un núcleo de borde solo transmitirá información sobre el borde de una imagen (Patterson y Gibson, 2017), que incluye el proceso de convolución que se muestra en la Figura 28:

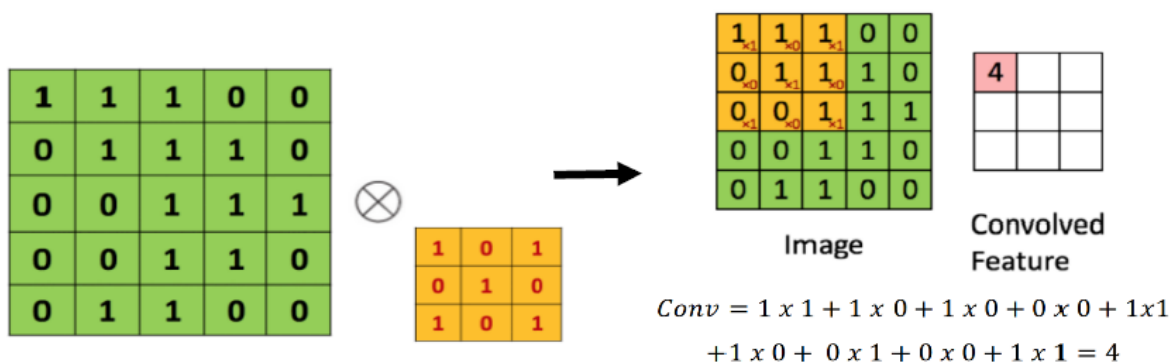


Fig 27 Operación de Convolución

Fuente: (Cicero, 2018)

2.4.2.1.3. *Función de Activación*

Al final del proceso de convolución, se obtiene como resultado una función lineal. Si desea conectar varias capas de neuronas cuyas salidas son funciones lineales, esto hace que la red se colapse de modo que sea equivalente a tener una sola neurona, lo que le impide trabajar en varias redes. Para resolver este problema y crear una arquitectura de red neuronal compleja, se utilizan funciones de activación para introducir la no linealidad en ConvNet porque la mayor parte de la información del mundo real que la red tiene que aprender no es lineal. (Enyinna et al., 2018). La función de activación comúnmente utilizada en CNN es ReLU (tasa lineal) y es igual a $f(x) = \max(0, x)$. Esta función reemplaza los valores negativos de entrada por cero, eliminando efectivamente la relación proporcional entre entrada y salida (Merchan et al., 2018) y reduciendo a cero la cantidad de estímulos generados por la salida, lo que aumenta la velocidad y mejora la convergencia de la red. Porque solo las características activas pasan a la siguiente capa (Canziani et al., 2016). A diferencia de otras funciones de activación, ReLU reduce el desvanecimiento, lo que afecta el entrenamiento de la red, porque los pesos cambian de manera insignificante, lo que hace que el aprendizaje de la red se detenga en las primeras capas, lo que afecta al resto de la red. (Huang et al., 2018)

2.4.2.1.4. *Capa de Polling*

Además de las capas de acumulación descritas anteriormente, también existe una Polling o Polling layer, la cual se encarga de simplificar la información en la salida de la capa compuesta, es decir, reduce el tamaño del mapa de características para mantener solo las más relevantes. características, reduciendo los parámetros y por lo tanto el costo computacional (Nilsen, 2018), como se muestra en la Fig. 29. Cabe señalar que en esta capa no hay parámetros de actualización porque su única función es reducir el tamaño. Los tipos de técnicas de montaje más habituales son el montaje máximo y el montaje medio. El agrupamiento máximo es una forma común de agrupamiento, que extrae subarreglos del mapa de características de entrada, genera el valor máximo en cada subarreglo y descarta todos los demás valores. El agrupamiento promedio calcula el promedio de cada submatriz del mapa de características (Wang & Yang, 2019).

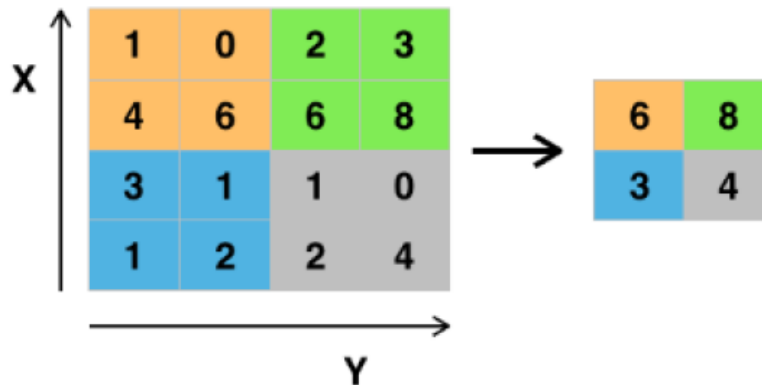


Fig 28 Ejemplo del funcionamiento de la capa de pool

Fuente: (Cicero, 2018)

La capa Polling asegura que la red neuronal tenga una propiedad llamada estabilidad espacial, es decir, independientemente de la presencia de distorsiones o cambios en la textura, iluminación, posición o rotación del objeto detectable, la red neuronal debería poder extraer características dominantes de ella, independientemente de las transformaciones resultantes (Yamashita et al., 2018). Esto es necesario porque los objetos se detectan independientemente de su posición u orientación en la imagen.

2.4.2.1.5. Capa Directamente Conectada

Después de una serie de capas convolucionales y de Polling, las características de nivel superior se extraen y pasan a capas directamente conectadas para realizar la tarea de clasificación basada en el conjunto de datos de entrenamiento. Por lo general, la cantidad de nodos conectados directamente a la salida de la capa es la misma que la cantidad de capas. La función softmax se utiliza para la tarea de clasificación multiclase, que normaliza los valores de salida de la clase final totalmente conectada, según la probabilidad de cada clase objetivo, donde cada valor va de 0 a 1 y la suma de estos valores es 1 (Yamashita et al., 2018).

Para que todas las capas trabajen juntas, se debe realizar un entrenamiento, que se resume en los siguientes pasos: Se asignan valores aleatorios a los parámetros o pesos de la red. El conjunto de datos de la imagen de entrenamiento se usa luego para ingresarlo en el modelo.

Se calcula el margen de error (función de pérdida) para las posibilidades obtenidas del modelo, luego se compara el resultado del cálculo con la salida deseada y finalmente se usa el método de retro propagación para calcular el error de gradiente para todos los pesos en la red y use el gradiente para cambiar estos valores repetidamente y reducir el error de salida (Merchan et al., 2018).

2.5. Lenguaje de Programación

Python es un lenguaje de programación interpretado de alto nivel y multiparadigma con escritura dinámica poderosa, administración automática de recursos, introspección avanzada y un sistema de manejo de excepciones. Es gratuito y gratuito, funciona en todas las plataformas, apareció en 1990 y tiene múltiples implementaciones, incluidas CPython, Jython, IronPython y PyPy. Su licencia es la "Licencia de la Fundación de Software de Python". Relativamente cercano a la licencia BSD y compatible con la licencia GPL. Su sintaxis es mínima, explícita, clara, simple y lo suficientemente cercana al lenguaje natural para que pueda comprender el algoritmo en la primera lectura. Una de las ventajas de este lenguaje es que el refinamiento de los reflejos (algoritmos compuestos por palabras) se reduce de forma prácticamente natural (Chacua Criollo, 2019)

2.6. Software de Entrenamiento y Procesamiento de imagen

2.6.1. Yolo

Solo se mira una vez (YOLO) es un sistema de detección de objetos en tiempo real de última generación. En un Pascal Titan X procesa imágenes a 30 FPS y tiene un mAP del 57,9% en COCO test-dev (Redmon & Farhadi, 2018). Se ejecuta significativamente más rápido que otros métodos de detección con un rendimiento comparable. Tiempos de un M40 o Titan X, son básicamente la misma GPU.

2.6.1.1. Yolo V3

YOLOV3 es extremadamente rápido y preciso. En mAP medido a .5 IOU YOLOv3 está a la par con la pérdida focal, pero aproximadamente 4 veces más rápido. Además, puede intercambiar fácilmente entre velocidad y precisión simplemente cambiando el tamaño del modelo, ¡sin necesidad de volver a entrenar! (Analogy et al., 2017)

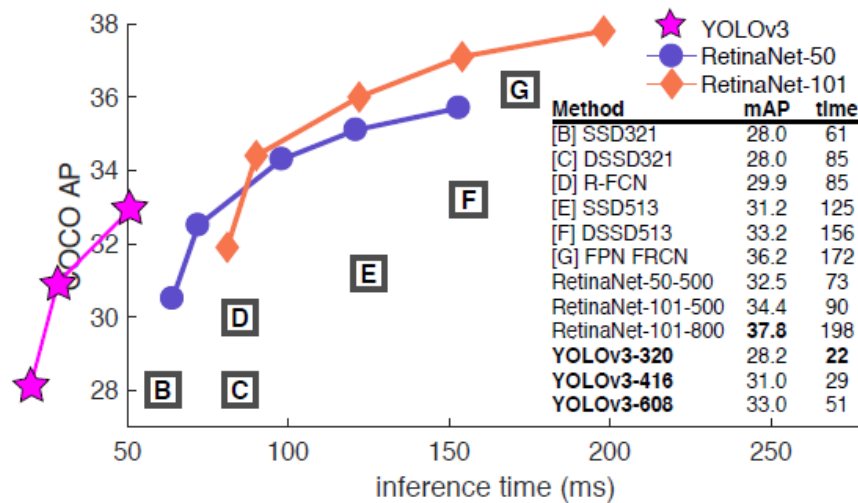


Fig 29 comparación de Yolo ante otros clasificadores

Fuente: (Goyal et al., 2018)

Los sistemas de detección anteriores mostrados en la Figura 30 reutilizan clasificadores o localizadores para realizar la detección. Aplican el modelo a una imagen en múltiples posiciones y escalas. Las regiones de puntuación alta en la imagen son las regiones de detección. (Redmon & Farhadi, 2018).

Usamos un enfoque completamente diferente. Aplicamos una única red neuronal a toda la imagen. Esta cuadrícula divide la imagen en regiones y predice cajas perimetrales y probabilidades para cada región. Estos cuadros delimitadores se calculan de acuerdo con las probabilidades esperadas (Analogy et al., 2017)

2.6.1.2. Tiny Yolo

Tiny Yolo es una versión comprimida de YOLOv3., se propone simplificar la estructura de la red y reducir los parámetros para hacer viable el desarrollo en dispositivos móviles y embebidos.

YOLOv3-tiny se puede utilizar para acelerar el entrenamiento y la detección. Tiene solo 2 cabezas YOLOv3-tiny en comparación con las 3 en YOLOv3 haciendo que esta se

entrene a partir de 29 capas convolucionales pre entrenadas en comparación con YOLOv3 que es entrenada a partir de 137 capas convolucionales pre entrenadas.

Los FPS (fotogramas por segundo) de YOLOv3-tiny es aproximadamente 8 veces mayor que el de YOLOv3; Sin embargo, cuando se prueba en el conjunto de datos MS COCO, la precisión de YOLOv3-tiny es 2/3 de la de YOLOv3

Para la detección de objetos en tiempo real. Esto se debe a que el tiempo de inferencia rápido es más importante que la exactitud y la precisión cuando se trabaja en un entorno de detección de objetos en tiempo real. (Khokhlov et al., 2020)

2.6.2. LabelImage

LabelImg es una herramienta de anotación de imágenes liviana y fácil de usar para etiquetar los cuadros delimitadores de objetos en las imágenes. Elegir el software de anotación de imágenes correcto es fundamental para el éxito a largo plazo de su aplicación de procesamiento de imágenes.

LabelImg es un programa de software gratuito y de código abierto para etiquetar imágenes con gráficos. Este software fue lanzado por Tzutalin en 2015 y está escrito en Python. Utiliza QT para GI (interfaz gráfica). LabelImg es una herramienta simple y básica para etiquetar cientos de imágenes con el fin de crear un conjunto de datos para entrenar un modelo de procesamiento de imágenes. Las anotaciones se pueden guardar como un archivo XML en formato PASCAL VOC. Pascal VOC es un formato de anotación XML popular utilizado por ImageNet que es legible por humanos, pero no funciona con modelos de detección de objetos conocidos. (*Labeling Images and Text Documents - Azure Machine Learning* | Microsoft Learn, n.d.)

2.6.3. Darknet

Darknet es un poderoso marco de código abierto para implementar redes neuronales. Escrito en C y CUDA, se puede integrar en CPU y GPU. Las implementaciones avanzadas de redes neuronales profundas pueden ejecutarse en la darknet. Estas implementaciones incluyen You Only Look Once (YOLO) para la detección de objetos en tiempo real, la

clasificación de ImageNet, las redes neuronales recurrentes (RNN) y muchas otras. (Redmon, 2013)

Darknet es un motor de red neuronal convolucional que utiliza modelos compuestos, siendo el uso principal de este motor la clasificación en tiempo real o casi en tiempo real de objetos en transmisiones de video, como archivos de video y cámaras. Escrito principalmente en el lenguaje de programación C, originalmente se ejecutó en GNU/Linux. Se puede usar en CPU más lentas, pero si es necesario, puede acelerarlo en GPU basadas en NVIDIA usando el kit de herramientas CUDA y agregando cuDNN al Tensors ComputeToolkit. (Lin et al., 2014)

Modelo	Top-1	Top-5	Operaciones	Tamaño
AlexNet	57.0	80.3	2.27 millones	238 MB
Referencia de Darknet	61.1	83.0	0.81 millones	28 MB
SqueezeNet	57.5	80.3	2.17 millones	4,8 MB
Tiny Darknet	58.7	81.7	0.98 millones	4,0 MB

Fig 30 Modelo de Tiny Yolo

3. DESARROLLO EXPERIMENTAL

Desarrollo Experimental

En el presente capítulo se va a presentar un análisis de la situación actual de los cultivos Micro-Farms al realizar la tabulación de la encuesta realizada al dueño de este en la ubicación del anexo 4, así como también una explicación de como estas preguntas ayudaron al desarrollo de un sistema de Visión Artificial (AI) a través de una plataforma para detectar las enfermedades. Por consiguiente, se procede a realizar una descripción del sistema donde se aclaramos los beneficios y posibles limitaciones, así como presentar el análisis de requerimientos de stakeholders, funcionalidades y arquitecturas, en base a la metodología Action-Reserch.

3.1. Metodología

En el desarrollo del presente trabajo de Investigación, se presenta como sugerencia una metodología de desarrollo de Action-Reserch, la cual cumple y se adapta de mejor manera al proceso de la creación del proyecto, investigación, diseño, codificación, pruebas y mantenimientos de software.

3.1.1. Action-Reserch

De esta manera se consigue una metodología práctica y flexible en la cual se aplica una investigación activa en la práctica sobre la elaboración de teorías o conocimientos en base a la acción y reacción (causa y efecto). Por lo tanto, en la metodología de action-reserch el investigador tiene como referencia la intervención del mundo real y obtener información o generar nuevas teorías o conocimientos en particular. (Mckay & Marshall, 2002)

3.1.2. Ciclo de Indagación Action-Reserch

Es importante reconocer la action-reserch como uno de los muchos tipos diferentes de Investigación de acción es un término colectivo para los procesos que siguen un ciclo de mejora de la práctica al alternar sistemáticamente la acción y el cuestionamiento basados en la práctica. O en planificar, implementa, describe y evalúa los cambios de mejora en su propia práctica mientras se aprende más sobre la práctica de la investigación en el proceso de action-reserch utilizando las 4 fases del ciclo básico de la investigación las cuales se muestran a continuación en la figura 31. (Tripp, 2005)

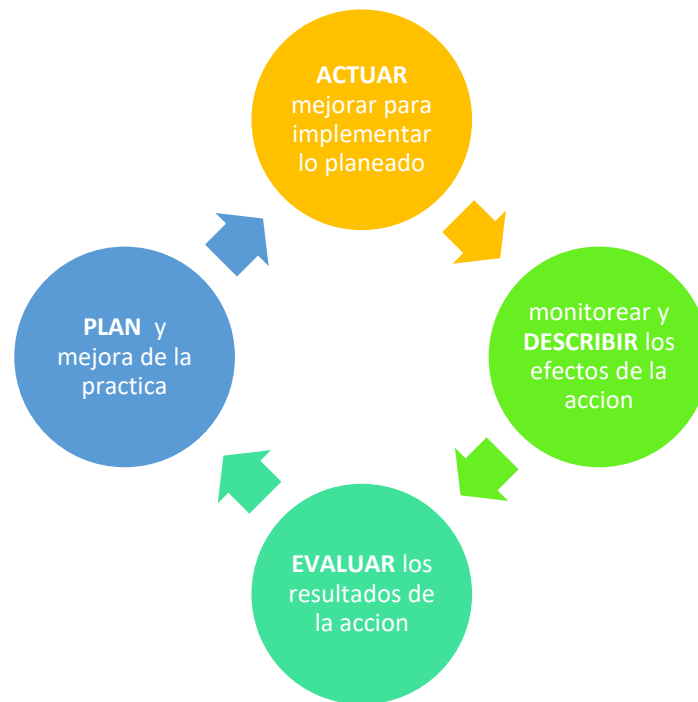


Fig 31 Fases del ciclo de la investigación

Fuente: (Tripp, 2005)

En el proceso circular de la investigación se la inicia en un proceso de planificación para el desarrollo de un sistema de inteligencia artificial con procesamiento de imagen en cultivos Micro-Farms, con el uso de sistemas de monitorización en tiempo real y un software de redes neuronales donde se describe la selección de cultivos hortícolas a estudiar.

La siguiente etapa es la de identificación de los ecos de diseñar una propuesta y presentar la selección tanto en hardware y de software a utilizar, en base a los BenchMark que se basarían tanto en los requerimientos del presente proyecto como en las herramientas que se adapten mejor al entorno en el cual se va a trabajar.

La etapa de desarrollo y análisis, esquematizando el sistema a usar basando en redes neuronales y la integración que llevara con los requerimiento tanto de hardware como de software y la implementación de la misma en cultivos hortícolas micro-farms, permitiendo la adquisición de datos y variables de imagen con buena resolución y precisión, adicional a

ello, obtener un enfoque multidisciplinario evaluando el desempeño y costos computacionales a la hora de la ejecución del algoritmo de aprendizaje de máquina .

En la etapa consecutivo, la implementación del software es importante en cuanto a la obtención de las variables de entre las cuales se tomaron en cuenta la pigmentación el crecimiento y la enfermedad o virus con la finalidad de etiquetar los resultados, finalizando con los resultados finales a la hora de entregar o visualizar un reporte en tiempo real de las características del cultivo hortícola verificando el sano crecimiento de este.

En resumen, la metodología Action-Reserch, permite comunicar y evaluar los procesos relevantes de manera fundamental, con el objetivo de mejorar las condiciones existentes en el tema abordado, en beneficio del cultivo y la inversión. Donde las etapas transforman el conocimiento de las maquinarias de tecnología avanzada y las actividades hortícolas convencionales.

3.2. Análisis

A través de este tipo de análisis lo que se busca es recabar toda la información posible para poder tener referentes acerca del diseño tanto en hardware como en software, por lo cual el apoyo de toda la información recabada directamente del dueño del cultivo Micro-Farm, serán la base para la investigación que se realiza, presentado así una solución a la problemática planteada.

3.2.1. Situación Actual

En la actualidad en base a los datos recabados en cuento al cultivo Micro-Farm en donde se están realizando las pruebas de funcionamiento nos arrojan como resultados errores drásticos para el hardware del sistema causando sobrecalentamientos al procesador de la placa principal “Raspberry” debido a la falta de procesamiento en el dispositivo por lo cual, se ha realizado un cambio estratégico del hardware para poder obtener un mejor resultado sumando más procesamiento al previsto ayudando al Neural Stick II de Microsoft, teniendo mejor respuesta en el tiempo de procesamientos.

Con el problema antes mencionado resuelto se puede obtener un sistema de mejor calidad para el usuario final teniendo mejores adaptaciones al cultivo propuesto, así como resultados más eficaces y eficientes al momento del reconocimiento de imágenes.

Para realizar el sistema propuesto, se ha considerado los cultivos micro-farms de lechuga mencionados a continuación: Cultivo de lechuga “THE CREEPS” ubicado en el sector del mercado mayorista, el mismo que favoreció la recolección de datos, así como la realización de las pruebas respectivamente

El proyecto se lo realizó en los cultivos Micro-Farms a cargo del Ingeniero Ronni Andrade el cual lo ha llevado al ámbito microempresarial conjunto con su socia la Señorita Saray quienes actualmente poseen un sembrío de lechugas crespas las cuales son de mayor comercialización y de menor tiempo de crecimiento en cuanto al cultivo Micro-Farm dentro de su domicilio obteniendo mejores resultados que la agricultura convencional ya que llevan un control más minucioso en cuanto a su cultivo

En la actualidad los cultivos crecen sin mayor problema, sin embargo; en dichos cultivos se ha empezado a notar un tipo de enfermedad “Erysiphe cichoracearum” la cual es causada por excesivo consumo de nutrientes y de agua lo cual ocasiona que la lechuga se torne de un tono blanquecino mostrando síntomas de dicha enfermedad por lo cual se le presentó este sistema de visión por computador ayudado a que los cultivos tengan un mejor crecimiento controlando esta enfermedad a tiempo y evitando que las demás lechugas contraigan la enfermedad

El cultivo de lechugas Micro-Farm que posee el Ingeniero Ronni Andrade mantiene un control automatizado del riego el cual produce que en la etapa vegetativa en la cual la planta ha superado su etapa de plántula y comienza el brote de sus primeras hojas “verdaderas” en esta etapa la lechuga ha comenzado a tener los primeros brotes de la enfermedad “Erysiphe cichoracearum” haciendo que las hojas se tornen del color blanco de aspecto pulverulento. Las hojas se tornan amarillas y luego se secan. No suelen causar la

muerte de la planta, pero la debilitan y necrosan a su vez puede contagiar al resto del cultivo haciendo que la perdida de lechugas sea mucho mayor.

Con lo mencionado anteriormente se propuso un sistema de visión artificial el cual controlaría mediante el procesamiento de imágenes el aparecimiento de la enfermedad “Erysiphe cichoracearum” teniendo un control más detallado así como controlar de mejor manera el riego y los nutrientes de las lechugas, evitando con anticipación que esta enfermedad pueda generarse a futuros sembríos, ayudando también, al cultivo y su crecimiento especialmente en la etapa vegetativa donde presenta mayor aparición de dicha enfermedad



Fig 32 Ubicación Geográfica The Creeps

Para la obtención de algunos de los datos se utilizaros métodos de encuestas como objetivo de análisis, en la cual se quiere conocer las razones reales del desarrollo del cultivo y como se vería afectado al implementar un sistema de visión artificial, además; con el tipo de preguntas cerradas las respuestas nos arrojan resultados con los cuales obtenemos la cuantificación y la tabulación de estas, las preguntas fueron referidas en cuanto al tiempo de

crecimiento, métodos de riego y tipos de enfermedades o virus que se pueden generar debido al tiempo del riego.

3.3.Introducción al desarrollo del proyecto

En el procesamiento y temas afines al desarrollo de sistemas de Inteligencia Artificial se han posesionado como una de las técnicas más solicitadas y remuneradas en cuanto a un sinnúmero de aplicaciones agrícolas para la identificación de virus y enfermedades de una manera mucho más efectiva y eficaz, obteniendo así un resultado mucho más significativo en factores de producción para beneficios económicos.

3.3.1. Propósito del Sistema

En la propuesta a desarrollar presenta como principal objetivo fundamental en el que sustenta gran parte del proyecto es diseñar una solución adecuada al problema de la detección de enfermedades (*Erysiphe cichoracearum*) en cultivos Micro-Farms, bajo el esquema de visión artificial, con lo cual ayudara a la mejora de la capacidad de las personas con cultivos pequeños en sus hogares para la implementación de medidas preventivas en sus cultivos de lechugas. Esta información se la reflejara en tiempo real cada cierto tiempo mejorando la determinación de la presencia de este tipo de enfermedad en los cultivos

3.3.2. Características de los Beneficiarios

Los propietarios de “THE CREEPS” son considerados beneficiarios y el sistema está desarrollado para realizar operaciones para el sector el tipo Micro-Farms. El sistema propuesto tiene características fáciles de usar. En este caso, los propietarios utilizan una plataforma dentro del computador que les permite ingresar de manera intuitiva, para que puedan visualizar los resultados del sistema de una manera clara y comprensible con la ejecución de un Script.

Se espera que los usuarios puedan interpretar esta información de forma gráfica y en tiempo real para eliminar rápidamente la enfermedad *Erysiphe cichoracearum* en los cultivos de lechugas

3.3.3. **Ámbito del Sistema**

En el sistema de visión por computador puede identificar la enfermedad (*Erysiphe cichoracearum*) en los pequeños cultivos Micro-Farms con un esquema de mejora en la adquisición y procesamiento de imágenes a otro nivel de laboratorio, lo que significa que las pruebas proporcionarían información adecuada para futuras implementaciones dentro de la industria Micro-Farm

Tenga en cuenta que este sistema es rentable y está destinado a contribuir al ahorro económico de los agricultores y o personas particulares que gustan de este tipo de cultivos. El concepto mejora la productividad de los cultivos al detectar anomalías y permitir que los usuarios intervengan rápidamente con soluciones eficaces.

3.3.4. **Dimensionamiento de Stakeholders**

En primera instancia, se dimensionan los Stakeholders implicados, directa o indirectamente que se necesitarían para el desarrollo del sistema; así como detallar los Stakeholders que promueven un papel importante con distintos grados de compromiso y responsabilidad, quienes buscan la manera eficaz para que el proyecto termine satisfactoriamente.

Tabla 9 Stakeholders

Stakeholders	
Usuarios Directos	
Administrador	Ing. Ronni Andrade
Director	Msc. Luis Suarez
Asesor	Msc Jaime Michilena
Desarrollador	Sr. Jorge Criollo

Nota Fuente: Autoría Propia

3.3.5. **Técnicas de Recolección de Información**

Las técnicas de recolección de información son todas las actividades y procedimientos que permiten a los investigadores el acceso directo a toda la información

necesaria, logrando así los objetivos de investigación necesarios y los requisitos habilitadores (Moncayo Suárez, 2021)

Se mencionan las siguientes técnicas de recolección de información:

- Observación: En esta técnica se extrae de forma metódica y a través de la revisión, los datos.
- Recopilación documental: En esta técnica se obtiene datos e información a partir de fuentes documentales con el fin de ser utilizados.
- Entrevista: Es una técnica para recabar datos; se define como una conversación que se propone un fin determinado distinto al simple hecho de conversar. Es un instrumento técnico que adopte la forma de un diálogo.
- Encuesta: Es una técnica de recopilación de información donde el investigador interroga a los investigados los datos que desea obtener ya sea mediante cuestionarios, preguntas, etc.

Para este proyecto se emplean las técnicas de observación, la entrevista y la encuesta; las cuales servirán de herramientas para tener información sobre el sistema a realizarse.

3.4.Requerimientos

Se requieren algunas cosas para que el proyecto funcione correctamente. Los requisitos cumplen con la última actualización del estándar ISO/IEC/IEEE 29148:2018 (“ISO/IEC/IEEE,” 2018) A lo largo del ciclo de vida del proyecto, incluye no solo proyectos de sistemas artificiales, sino también productos de hardware de software, servicios y productos relacionados con el sistema. Las preferencias del usuario se mantienen en relación con los requisitos del proyecto, centrándose en el diseño, la implementación y el rendimiento del sistema.

3.4.1. Nomenclatura de los Requerimientos

A continuación, en la Tabla 11 se detalla las abreviaturas utilizadas para una mejor comprensión y fluidez de términos en el texto, así como de la información acorde a los

requerimientos necesarios del sistema, requerimientos de los involucrados, así como los requerimientos de la arquitectura

Tabla 10 Términos Abreviados y Acrónimos Usados Durante el Análisis de Requerimientos

Acrónimos y Abreviados	Descripción
StSR	Requerimientos de Stakeholders
SySR	Requerimientos del Sistema
SRSR	Requerimientos de Hardware y Software

Nota Fuente: (“ISO/IEC/IEEE,” 2018)

3.4.2. Requerimientos de Stakeholders

Para el desarrollo del proyecto se establecen las especificaciones técnicas de requerimientos en base a una mesa de trabajo desarrollada por el Lic. Luis Suárez, Especialista Técnico en Sistemas Inteligentes. Como resultado de nuestro trabajo, contamos con las siguientes especificaciones: Una prioridad que se clasifica entre alta, media y baja según la importancia de los requisitos del sistema, las precauciones y el Requisito (consulte la Tabla 11).

Tabla 11 Priorización de los Requerimientos para el Sistema

Prioridad	Descripción
Alta	Considera un nivel de suma importancia para el excelente funcionamiento del sistema.
Media	Estos requisitos se pueden omitir en condiciones extremas, pero no incluirlos afectarían parcialmente al sistema.
Baja	Pueden ser omitidos, no repercuten un impacto en el correcto funcionamiento del sistema.

Nota Fuente: (Andrade Guevara, 2022)

En si la finalidad por la cual se usan estos requerimientos es definir los requisitos adecuado para el sistema basándose en las necesidades del usuario final en el campo de cultivos Micro-Farms. Por lo cual, para poder cumplir con las expectativas de los

propietarios, por lo cual en la tabla 13 se evalúan los requerimientos operacionales y de usuario

Tabla 12 Requerimientos del Propietario

#	REQUERIMINETOS	PRIORIDAD			RELACION
		ALTA	MEDIA	BAJA	
REQUERIMIENTOS sistema					
STRS1	Disponibilidad de una toma de corriente eléctrica, para la alimentación del computador	X			
STRS2	Disponibilidad de conectividad USB o Wifi para el funcionamiento de la cámara	X			
STRS3	Disponibilidad de un lugar estratégico para la ubicación de la cámara	X			
STRS4	Disponibilidad de un lugar adecuado para el monitor que estará mostrando la información	X			
STRS5	El sistema debe tener la capacidad de funcionar durante 1 hora por semana	X			
REQUERIMIENTOPS DE USUARIO					
STRS6	El acceso a la información y al control master de la aplicación será mediante autenticación de usuario y password	X			
STRS7	Los resultados y alertas se mostrarán de manera inmediata en el monitor cada hora por semana	X			
STRS8	La cámara debe ubicarse de tal manera que el sistema pueda realizar el reconocimiento de la enfermedad a escala de laboratorio	X			
STRS9					
STRS10					

Los requerimientos planteados han sido obtenidos en base a una entrevista con el dueño del cultivo Micro-Farms, así como una encuesta al mismo y su socio, además, se llegó a considerar la experticia para la ejecución de este tipo de proyecto, de tal manera conseguir una relación entre el usuario y el sistema

3.4.3. Requerimientos del Sistema

Los requisitos se realizan de acuerdo con las capacidades y limitaciones que realiza el sistema de discretización de imágenes. Debajo, analizamos el uso, el rendimiento, la interfaz, el estado y los requisitos físicos de acuerdo con los requisitos de las partes interesadas. La Tabla 14 contiene información sobre los requisitos del sistema. Este campo indica las capacidades del sistema en base a la aplicación según la prioridad de los requerimientos. Estos están relacionados con los requisitos de las partes interesadas.

Tabla 13 Requerimientos del Sistema

STSR					
REQUERIMIENTOS DEL SISTEMA					
#	REQUERIMINETOS	PRIORIDAD			RELACION
		ALTA	MEDIA	BAJA	
REQUERIMIENTOS DE ITERFAZ					
STRS1	Conectividad a tecnología de comunicación inalámbrica	X			
STRS2	Almacenamiento mínimo de 250 GB	X			
STRS3	Disponibilidad de conectividad de cámara USB o IP	X			
STRS4	La cámara debe tener la capacidad de capturar las diferentes tonalidades de color de la hoja del cultivo para la posible detección de enfermedades	X			
STRS5	El computador debe tener la capacidad de procesamiento de imágenes para tener un buen resultado final	X			
REQUERIMIENTOPS DE USO					
STRS6	Los datos deben ser obtenido de manera rápida y precisa	X			
STRS7	Los procesos deben rendir a largo plazo y no saturarse	X			
STRS8	La cámara debe tener la flexibilidad de la toma de datos de principio a fin	X			
REQUERIMIENTOS DE ESTADO					
STRS9	El computador debe estar encendido para el uso de software	X			
STRS10	La cámara debe estar encendida y funcional para realizar el reconocimiento de la enfermedad Erysiphe cichoracerum	X			

STRS11	El sistema debe ejecutarse automáticamente para realizar el funcionamiento	X
REQUERIMIENTOS FÍSICOS		
STRS12	La cámara necesita estar en una zona sin obstáculos y de buena luminosidad para toma de datos	X
STRS13	El sistema debe tener capacidad suficiente de almacenamiento para evitar saturaciones o colapsos de sistema	X
STRS14	Se debe contar con el lugar adecuado para realizar el reconocimiento de la enfermedad Erysiphe cichoracerum	X
STRS15	Realizar las pruebas a escala de laboratorio	X

3.4.4. Requerimientos de Arquitectura

Los requisitos arquitectónicos se refieren a los componentes de hardware y software en cuyas características se basan en cómo debe operar el sistema de visión por computador. Para este propósito se emplean requisitos lógicos, de diseño, hardware, software y eléctricos. La Tabla 15 muestra los requisitos arquitectónicos.

Tabla 14 Requerimientos de la Arquitectura

STSR					
REQUERIMIENTOS DE ARQUITECTURA					
#	REQUERIMINETOS	PRIORIDAD			RELACION
		ALTA	MEDIA	BAJA	
REQUERIMIENTOS DE DISEÑO					
STRS 1	El sistema debe tener una compatibilidad con la cámara	X			
STRS 2	Comunicación con la plataforma de procesamiento de datos tanto darknet como yolo	X			
STRS 3	Los datos de video de la cámara deben ser compatibles en la plataforma de yolo	X			

STRS 4	El sistema debe ser amigable con el usuario para el buen uso del administrador	X
STRS 5		
REQUERIMIENTOS LÓGICO		
STRS 6	Aplicación para poder conectar el móvil con el software en ejecución	
STRS 7	Acceso al software para muestra de resultados en tiempo real	
REQUERIMIENTOS DE HARDWARE		
STRS 8	La cámara debe ser como min de 180*720	
STRS 9	Conexión entre la cámara y el computador	
STRS	Capacidad de procesamiento de datos de alta velocidad	
STRS	Usar un procesador capaz de reconocer las imágenes en la realización del sistema	
REQUERIMIENTOS DE SOFTWARE		
STRS	Debe ser compatible con GNU/Linux o sistemas Open Source	X
STRS	Debe soportar librerías extensas openCV y Tensor Flow	X
STRS	Debe ser óptimo en el consumo de la RAM, para no sobrecargar el sistema	X
STRS	Soporte en tiempo real	X
STRS	Debe ser gratuito	X
REQUERIMIENTOS ELECTRICOS		
STRS	Disponer de tomacorriente para la conexión del computador	X
STRS	Disposición de fuente UPS	X
STRS		

3.5.Recursos

Es muy importante el tener en consideración que para la elaboración e implementación del proyecto se tiene que integrar recursos que son considerados de suma importancia para poder alcanzar los objetivos planteados. Dicho recurso se lo plantea de la siguiente manera.

3.5.1. Recursos económicos

Permite evaluar diferentes tipos de costos aproximados para el desarrollo del proyecto final, este tipo de gastos que se generan para el diseño del proyecto sumando los recursos administrativos para la presentación de este se muestran a en la tabla 16

Tabla 15 Tabla de Recursos

RECURSOS ECONOMICOS				
#	MATERIALES	COSTOS U.	# DE ELEMENTOS	COSTOS T.
1	Laptop	\$ 1450	1	1450.00
2	Resmas de Hojas	\$ 5	1	5.00
3	Empastados	\$ 38	1	35,00
4	Neural Stick II	\$ 165	1	165.00
5		\$	1	

3.5.2. Recursos tecnológicos

En este apartado se definen los posibles aparatos tecnológicos los cuales vamos a usar para el desarrollo del prototipo, de los cuales se detallan en la tabla 17.

Tabla 16 Requerimientos Tecnológicos

STSR					
REQUERIMIENTOS OPERACIONALES					
#	REQUERIMINETOS	PRIORIDAD			RELACION
		ALTA	MEDIA	BAJA	
REQUERIMIENTOS					
STRS1	Monitores	X			
STRS2	Neural Stick II	X			
STRS3	Camara	X			
STRS4	Computador	X			

3.6. Descripción General del Sistema

El sistema de visión por computador se encuentra compuesto por varios elementos de los cuales tendremos una pantalla para la visualización los resultados del algoritmo en el

reconocimiento de *Erysiphe cichoracearum* la pantalla se encuentra conectada al computador en el cual se realizan todos los procesamiento de imagen y donde se realiza el entrenamiento de las redes neuronales previas a la ejecución del sistema

Para la realización de todo el funcionamiento del sistema, se adquirieron fotografías de la enfermedad *Erysiphe cichoracearum* en los cultivos Micro-Farms de lechuga para el entrenamiento respectivo. Para las pruebas realizadas del sistema a escala de laboratorio la cámara adquirida se conecta directamente al computador para identificar la enfermedad a través de un entrenamiento mediante fotografías de la misma y poder mostrar buenos resultados al momento de ejecutarlo en tiempo real.

3.7. Elección de Hardware y Software

Seleccionar el hardware y el software adecuados implica varias formas en las que se pueden seleccionar los elementos ideales de forma técnica. Por lo tanto, al seleccionar el hardware, se realizan BenchMark utilizando hojas de datos de cada elemento, y al seleccionar el software, se integran los requisitos de las partes interesadas, los sistemas y las arquitecturas y se realiza un análisis comparativo para evaluar cada uno. Algo que puede evaluar para obtener opciones viables para la implementación de su proyecto.

3.7.1. Elección de Hardware

Para realizar el benchmarking, primero es necesario establecer algunas métricas a evaluar, las cuales se eligen en la Sección pertinente del Capítulo 2, que se enfoca en las aplicaciones agrícolas y de vigilancia, se seleccionaron este tipo de drones en base a su disponibilidad y precio en el mercado ecuatoriano, además de su compatibilidad con aplicaciones que realizan detección de imágenes.

3.7.1.1. Selección de Computador

A continuación, se realizará un benchmarking, el cual consiste en la comparación de características de los diferentes tipos de software que impacten entre los principales encontramos los mencionados a continuación en la tabla 18

Tabla 17 Selección de computador

Modelo	Imagen	Pantalla	Procesador	RAM	Disco Duro	Tarjeta gráfica	Precio
ASUS ZenBook 14 UM425IA-HM101T		14" Full HD. 1920 x 1080 píxeles.	AMD Ryzen 7 4700U	16 GB DDR4	512 GB SSD	AMD Radeon Graphics	1600
ASUS VivoBook 15 K513EA-BQ684		14" Full HD. 1920 x 1080 píxeles.	Intel Core i7-1065G7	16G DDR4	1 TB SSD	Intel Iris Plus Graphics	1690
Asus ROG Strix G15 Electro Punk Pink		15.6" Full HD. 1920 x 1080 píxeles.	i7-10750H CPU @ 2.60GHz 2.59 GHz	16 GB DDR4	512 TB SSD	NVIDIA GeForce GTX1650Ti	1450
ASUS ZenBook Pro Duo 15		15.6" Ultra HD. 3840 x 2160 píxeles.	Intel Core i7-10870H	32 GB DDR4	1 TB SSD	NVIDIA GeForce RTX3070	1565

Una vez realizada a tabla comparativa con diferentes características entre los computadores mencionados anteriormente, se llega a considerar que por las mejores características en cuanto a relación precio y con mejores características computacionales el asus rog strix g15 electro punk Pink tiene mejores prestaciones para el desarrollo del proyecto en cuanto a una ram de 16 gb ddr4 así como una tarjeta gráfica Nvidia Gforce GTX 1650ti la cual tiene compatibilidad con la plataforma de ejecución de darknet así como un procesador adecuado para el procesamiento de imágenes

3.7.1.2. Selección de Acelerador de GPU

De la misma manera realizamos el benchmarking en cuanto a nuestra selección de acelerador de procesamiento los cuales colocamos los más importantes y los más eficientes en cuanto al desarrollo del proyecto presentándolos en la tabla 19 a continuación

Tabla 18 Selección de Acelerador de GPU

Modelo	Acelerador USB Coral	Intel Movidius Neural Compute Stick2 w/Myramid
Imagen	-	-
Valoraciones	-	-
Opiniones	Prime	-
Prime	Unidad de procesamiento de tensor (TPU)	Unidad de procesamiento de visión (VPU)
Acelerador/Procesador	Unidad de procesamiento de tensor (TPU)	Unidad de procesamiento de visión (VPU)
Conexión	Puerto y cable USB 3.1	USB 3.0 tipo A
Dimensiones	30 x 65 x 8 mm	72,5 x 27 x 14 mm
Velocidad	4 TOPS	4 TOPS
Sistema Operativo	Linux Debian 10 (como Ubuntu 18.04), Armv7 (32 bits) o Armv8 (64 bits), MacOS 10.15, con MacPorts o Homebrew instalados y Windows 10.	Ubuntu, CentOS, Windows 10 y otros sistemas operativos
Modelos admitidos	TensorFlow Lite	TensorFlow, Caffe, ApacheMXNet, Open Neural Network Exchange, PyTorch y PaddlePadle

Para la selección del dispositivo de aceleración de procesamientos de la misma manera hemos decidido optar por un Intel neural stick 2 ya que en comparativas principales es más adecuado con el proyecto a realizarse ya que mantiene una gran cantidad de modelos de entrenamiento, así como librerías y compatibilidad de la misma manera con el sistema operativo a utilizarse y la velocidad de ejecución que tiene.

3.7.2. Elección de Software

Basándose en los requerimientos de software anteriormente mencionados en la tabla 15 se procede a la selección del software más adecuado para la realización del proyecto

3.7.2.1. Lenguaje de programación

Después de detallar los requisitos para elegir un lenguaje de programación, explicaremos cada requisito y decidiremos qué opción es adecuada para desarrollar el proyecto. Todas las características del lenguaje de programación los cuales serán útiles para el análisis de requisitos.

Para dejar espacio para la elección de lenguajes de programación, la Tabla 20 presenta un análisis que considera los requisitos de software.

Tabla 19 Selección de Lenguaje de Programación

Requerimientos	Python	Matlab
SRSH8	1	0
SRSH9	1	1
SRSH10	1	1
SRSH11	1	0
SRSH12	1	0
Valoración	5	2
Cumplimiento	0 – No Cumple 1 – Si Cumple	

En base al análisis descrito en la Tabla 20 y una vez descritas las características en los Anexos, en el presente proyecto se escogió una programación en Python; ya que, gracias a su gran variedad de librerías que facilitan la programación de clasificación gracias a que mantiene un soporte para el uso de tiempo real, así como ser un Software libre a diferencia de Matlab que es más utilizado para proyectos más pesados, de la misma manera no majo librerías adecuadas y completas usándole en su modo free.

3.7.2.2. Software de Tratamiento de Imágenes

En esta etapa seleccionaremos un algoritmo de clasificación de imágenes y que nos ayude de la misma manera en la detección de objetos deseados; es decir, lo que se realiza con

el algoritmo es la parte de la detección (los cuadros por pixeles en cuanto a x e y, así como el alto y ancho)

Por lo tanto, se ha desarrollado una tabla comparativa en la cual se detallan las caracterices mencionadas con anterioridad los cuales nos ayudaran para el análisis de los requerimientos en la tabla 21 a continuación.

Tabla 20 Software de Tratamiento de Imagen

Requerimientos	Yolov3	YoloV3 Tiny	Yolov4	YoloV4 Tiny
SRSH8	1	1	1	1
SRSH9	1	1	1	1
SRSH10	0	1	0	1
SRSH11	1	1	1	1
SRSH12	1	1	0	1
Valoración	4	5	3	5
Cumplimiento	0 – No Cumple 1 – Si Cumple			

Del análisis comparativo se obtiene dos opciones las cuales las podemos ocupar para el desarrollo del proyecto de los cuales en cuanto a las versiones Tiny de YOLO manejan menos interacciones para obtener una precisión alta a la hora de la detección de objetos y que a su vez es más ligera en cuanto a la velocidad de inferencia.

Basándose en el análisis se puede obtener una referencia para el entrenamiento del algoritmo de procesamiento de imagen, puesto que estaría basándose en la estructura Tiny en su versión 3 debido a que es la más estable en cuanto al funcionamiento del reconocimiento de imagen, por consiguiente, se lo utiliza como referencia, de esta manera se crearía un script único y exclusivo para la detección de la enfermedad Erysiphe cichoracerum de la lechuga.

3.8. Diseño del Sistema

Una vez que se definen los requisitos de hardware y software, continúa el diseño del sistema. Considerando todos los criterios anteriores; los mismos que permiten el desarrollo e implementación del Sistema de Detección de la enfermedad Erysiphe cichoracerum de la Lechuga.

. En esta etapa, la funcionalidad del sistema se vuelve clara gráficamente. Sirve como guía para seguir correctamente el proceso.

3.8.1. Diagrama de Bloques del Sistema

En esta sección, creamos un diagrama de bloques del sistema de detección de la enfermedad (). En este diagrama, hay dos fases que constituyen el proceso de realización del sistema y su ejecución. En la implementación del sistema se establecen una serie de pasos que comienzan con la adquisición de imágenes a partir de datos tomados de la enfermedad asociado a la recolección de lechuga. A continuación, etiquete la enfermedad de la lechuga con el software gratuito LabelImg. El entrenamiento del sistema se realiza a través de una arquitectura CNN. Una vez que el sistema está funcionando, las características de la enfermedad se extraen en tiempo real, fotos o videos almacenados. Basado en entrenamiento previo para el reconocimiento.

La figura a continuación muestra un diagrama de bloques del sistema de detección de enfermedades en un cultivo Micro-Farms de lechugas, lo que permite una mejor comprensión del proceso anterior.

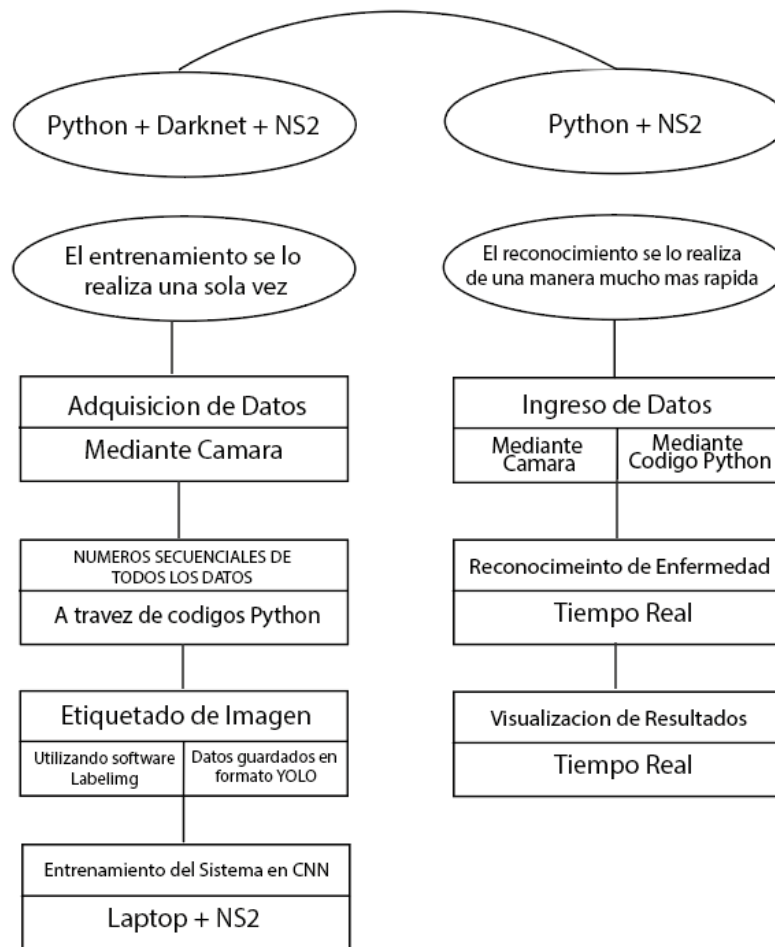


Fig 33 Diagrama de Bloques del sistema

Fuente: Autoría Propia

3.8.1.1. Diagrama del Funcionamiento del Sistema

Un diagrama de operación especifica el diseño del software para realizar un sistema de visión por computadora. En este punto, se entrega un diagrama del sistema que se divide en dos partes. El Cuadro No. 1 se muestra en la Figura 32

. Comienza con la recopilación de datos. Que consiste en la obtención de una serie de fotografías de la enfermedad Erysiphe cichoracerum en cultivos Micro-Farms de lechuga. A continuación, se etiqueto cada imagen con el software LabelImg (una herramienta de etiquetado de imágenes gráficas). da una imagen delimitada después del etiquetado. Los

datos anotados de salida se guardan en formato Yolo para compatibilidad con las implementaciones del sistema.

Luego, el sistema es entrenado por una red neuronal utilizando el lenguaje de programación de software libre Python. En este caso, se utilizan las herramientas de aceleración Neural Stick 2 de Microsoft que brinda una mayor cantidad de GPU y un procesamiento eficiente para el entrenamiento adecuado del sistema.

3.8.2. Diagrama de Flujo

Los diagramas de flujo se inicializan con código escrito en Python. Después de ingresar a la carpeta donde se almacena el código, se decide si usar las imágenes almacenadas o el video almacenado para realizar experimentos a escala de laboratorio en tiempo real. Cuando se selecciona una de las opciones, se ejecuta el código deseado para mostrar la tasa de detección obtenida.

La Figura 33 muestra un diagrama de flujo del sistema de detección de dicha enfermedad en cultivos Micro-Farms de lechuga, lo que permite una mejor comprensión del proceso anterior.

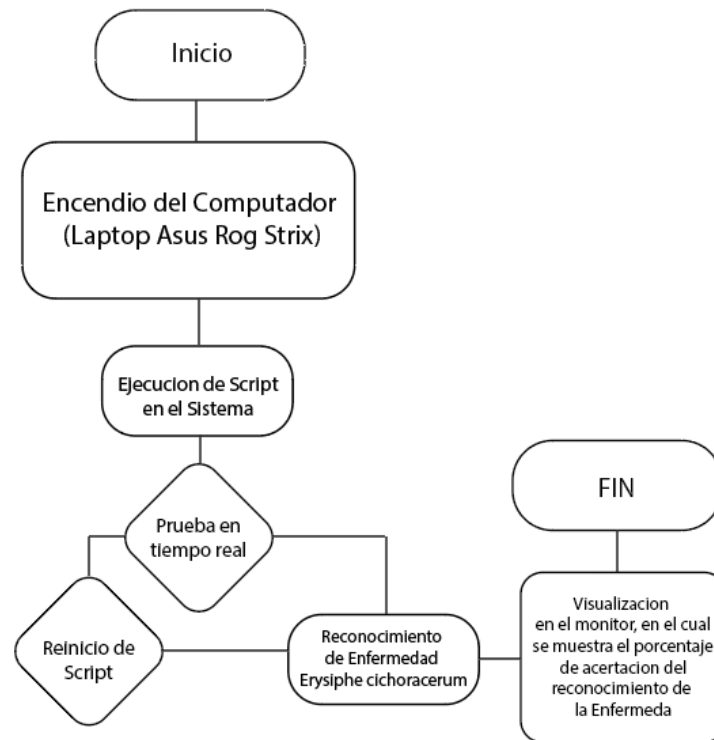


Fig 34 Diagrama de Flujo del Sistema

Fuente: Autoría Propia

3.8.3. Diagrama de la Arquitectura del sistema

El sistema primordialmente cuenta con un computador con un Sistema Operativo Linux que estaría operando como eje central del sistema, este se encuentra mostrando los resultados a la hora de la ejecución del software, de la misma manera se encuentra conectada a este una cámara la que está encargada de realizar el reconocimiento de la Enferma () en tiempo real.

En la figura 34 a continuación se observa el diagrama de la arquitectura de; sistema de reconocimiento en el cual consta a detalle el procedimiento que se realiza para obtener los resultados esperados, así como el correcto funcionamiento del mismo sistema.

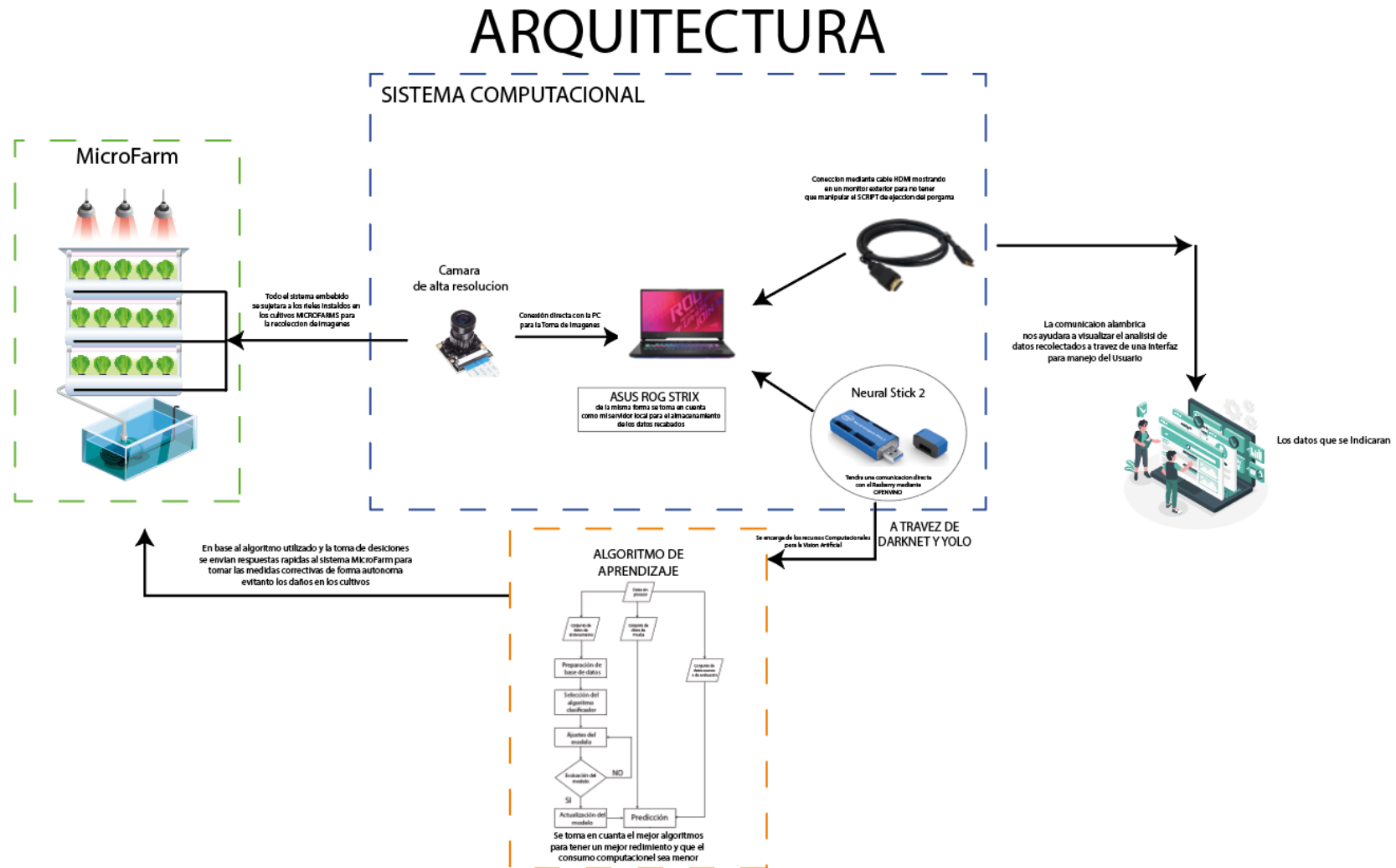


Fig 35 Arquitectura del Sistema

Fuente: Autoría Propia

3.8.4. Desarrollo del Software

En esta sección, analizamos el desarrollo de cada fase que tiene el software de programación en el proceso general de visión artificial. Su desarrollo comienza con la detección de la Enfermedad Erysiphe cichoracerum mediante el sistema Yolo Tiny V3. Cubriremos la configuración del sistema de detección de objetos Yolo Tiny V3, Darknet, cómo crear patrones de entrenamiento, cómo entrenar el modelo y finalmente cómo probar el modelo para detectar Enfermedades en tiempo real con modelos entrenados en Yolo Tiny V3.

3.8.4.1. Yolo

Yolo es una red convolucional única que predice conjuntamente las probabilidades de múltiples interfaces y múltiples clases. Yolo usa características de toda la imagen para predecir cada cuadro delimitador. De manera similar, para cada clase de imagen, predecimos todos los cuadros delimitadores simultáneamente. Yolo divide la imagen de entrada en una cuadrícula $S \times S$ (los tamaños se construyen de acuerdo con cada criterio).

La célula se encarga de detectar enfermedades. Yolo está capacitado para usar fotogramas completos, lo que optimiza el rendimiento de detección. Una red de detección tiene capas convolucionales seguidas de capas totalmente conectadas. Cambiar entre capas convolucionales reduce el espacio de características de la capa anterior.

3.8.4.2. Configuración Darknet en Yolo Tiny

Una vez instalado, Darknet se puede utilizar en cualquier sistema operativo como Windows, Mac OS y Linux. En este caso, el entrenamiento se lleva a cabo en el ordenador directamente conjunto con la aceleración del Neural Stick II, permitiendo la ejecución de códigos de Python para programación directamente. Para este proceso se utiliza una computadora asus con 16 GB de memoria ram y un procesador Intel Core i7 10ma generación adicional con la colaboración de Intel Movidius Neural Compute Stick2 el cual nos ayuda con más procesos al momento de ejecutar el entrenamiento.

Para la ejecución de Tiny Yolo es necesario cumplir con los requerimientos mínimos para el buen funcionamiento de su estructura, para lo cual la configuración necesaria mínima que se debería tener en cuenta es la de tener un buen procesador el cual nos ayudara tanto a la ejecución y reconocimiento de la imagen en batches y subdivisiones así también como la resolución de la imagen en cuadros por segundo para un optimo reconocimiento por pixel como se puede apreciar a continuación en la Fig 36.

```
[net]
# Testing
# batch=1
# subdivisions=1
# Training
batch=2
subdivisions=2
width=416
height=416
channels=3
momentum=0.9
decay=0.0005
angle=0
saturation = 1.5
exposure = 1.5
hue=.1

learning_rate=0.001
burn_in=1000
max_batches = 500200
policy=steps
steps=400000,450000
scales=.1,.1

[convolutional]
batch_normalize=1
filters=16
size=3
stride=1
pad=1
activation=leaky
```

Fig 36 Configuración Principal de Tiny YoloV3

Fuente: Autoría Propia

3.8.4.3. Adquisición de imágenes

Para implementar el sistema propuesto y entrenarlo con redes neuronales convolucionales, en un primer proceso se obtienen fotografías del cultivo de lechugas, considerando una base de datos que tenga plantas en este caso con la enfermedad de Erysiphe cichoracearum para entrenar correctamente el algoritmo.

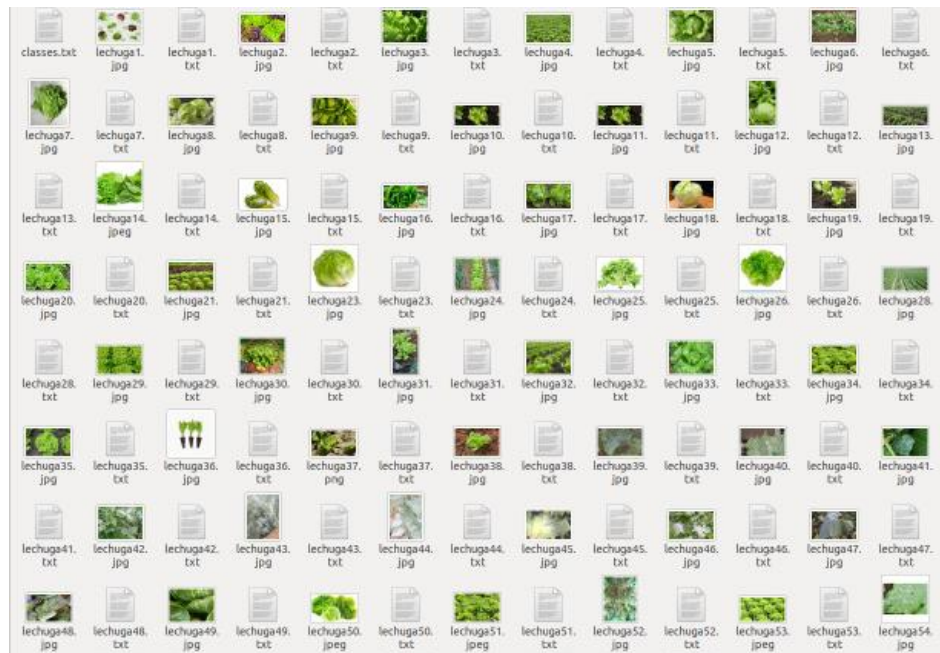


Fig 38 Base de datos en Orden

Fuente: Autoría Propia

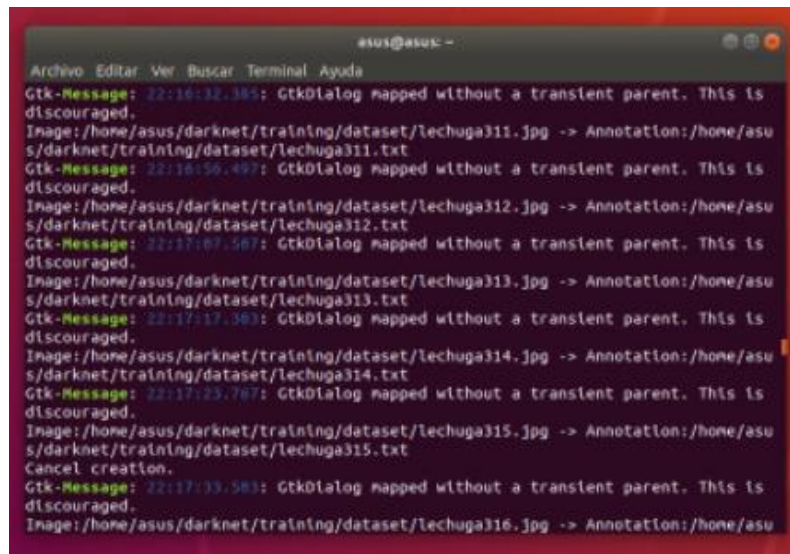
3.8.4.4. Etiquetado de imágenes

En este paso, las imágenes se etiquetan con el software LabelImg, una herramienta gratuita de fácil acceso y muy intuitiva de código abierto para etiquetar imágenes gráficamente directamente señalando el cuadro el cual parezca mucho más óptimo al momento de etiquetarlo para tener un mejor entrenamiento del algoritmo. LabelImg está escrito en Python, es una forma fácil y gratuita de etiquetar cientos de imágenes y admite el etiquetado en formatos de archivo de texto VOC XML o YOLO. Del cual se usará el de formato Yolo siendo el mismo que se utiliza para el desarrollo del proyecto en sí.

Para lograr un buen etiquetado de imágenes se toma a consideración 3 formas de identificar que se está realizando el etiquetado correcto, primeramente, se tiene a consideración el etiquetado de manera gráfica en la cual se realiza manual con la selección del recuadro que queremos seleccionar y etiquetar como se muestra a continuación en la fig 38.

Así también se debe tener en cuenta que adicional a esto y conjunto al etiquetado de manera gráfica se tiene también un archivo txt en el cual se crea el etiquetado de cada una de

las imágenes en la cual se visualizará las coordenadas exactas en los ejes x, y de cada una de las imágenes etiquetadas; de la misma manera en el terminal donde se ejecutó LabelImg se mostrara la dirección de donde se creó el documento así como el nombre que se lo tiene asignado de acuerdo al nombre de cada una de las imágenes que se toman para el etiquetado esto se visualiza a continuación en las fig 38 y 39 respectivamente.



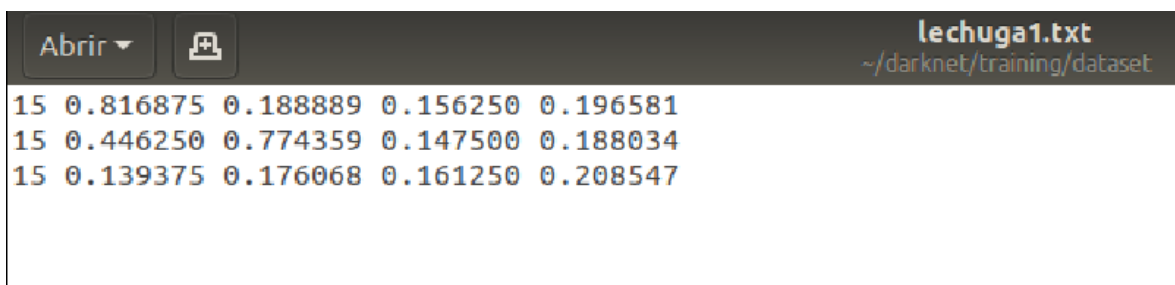
```

asus@asus: ~
Archivo Editar Ver Buscar Terminal Ayuda
Gtk-Message: 22:16:32.385: GtkDialog mapped without a transient parent. This is discouraged.
Image: /home/asus/darknet/training/dataset/lechuga311.jpg -> Annotation: /home/asus/darknet/training/dataset/lechuga311.txt
Gtk-Message: 22:16:50.497: GtkDialog mapped without a transient parent. This is discouraged.
Image: /home/asus/darknet/training/dataset/lechuga312.jpg -> Annotation: /home/asus/darknet/training/dataset/lechuga312.txt
Gtk-Message: 22:17:07.587: GtkDialog mapped without a transient parent. This is discouraged.
Image: /home/asus/darknet/training/dataset/lechuga313.jpg -> Annotation: /home/asus/darknet/training/dataset/lechuga313.txt
Gtk-Message: 22:17:17.383: GtkDialog mapped without a transient parent. This is discouraged.
Image: /home/asus/darknet/training/dataset/lechuga314.jpg -> Annotation: /home/asus/darknet/training/dataset/lechuga314.txt
Gtk-Message: 22:17:25.787: GtkDialog mapped without a transient parent. This is discouraged.
Image: /home/asus/darknet/training/dataset/lechuga315.jpg -> Annotation: /home/asus/darknet/training/dataset/lechuga315.txt
Cancel creation.
Gtk-Message: 22:17:33.583: GtkDialog mapped without a transient parent. This is discouraged.
Image: /home/asus/darknet/training/dataset/lechuga316.jpg -> Annotation: /home/asus/darknet/training/dataset/lechuga316.txt

```

Fig 39 Proceso de Etiquetado

Fuente: Autoría Propia



```

lechu1.txt
~/darknet/training/dataset
15 0.816875 0.188889 0.156250 0.196581
15 0.446250 0.774359 0.147500 0.188034
15 0.139375 0.176068 0.161250 0.208547

```

Fig 40 Resultado en Coordenadas del Etiquetado

Fuente: Autoría Propia

3.8.4.5. Entrenamiento del Algoritmo Mediante Redes Neuronales Convolucionales

Después de adquirir y etiquetar previamente los cultivos de lechuga con la herramienta LabelImg, los archivos de cada imagen se guardan directamente en la carpeta de Data Base. Los archivos producidos son del tipo (.txt) que contienen las coordenadas exactas del etiquetado de la imagen, mostrando la ubicación de identificación de la lechuga, así como de la enfermedad Erysiphe cichoracearum; Cabe señalar que el formato creado por la herramienta LabelImg guarda archivos en un formato compatible con Yolo.

Si tiene imágenes en el formato apropiado (.txt), debe colocarlas en la carpeta de training, esta carpeta se encuentra en la raíz de Darknet. Por otro lado, para crear un data set para entrenamiento, se deben crear dos carpetas, una que contenga las imágenes a detectar, así como las coordenadas y otra que contenga los modelos de programación para que logren detectar el número de convoluciones necesarias, así como el computador con toda la RAM disponible.

Este proceso ha terminado; Descarga tus escalas modelo de entrenamiento desde la página oficial de Darknet Tiny Yolo V3.

3.8.4.5.1. Parámetros de configuración de la red

Para tener un buen entrenamiento y configuración de la red neuronal, se tiene algunos puntos a tomar en cuenta para ello se los detalla a continuación cada uno.

- Cantidad de cajas o cuadros por imagen: en la cual se detallarían la cantidad de objetos máximos que queremos detectar.
- Epoch: aquí detalla la cantidad de interacciones sobre todos los datasets que se realizara en la red neuronal a partir de su entrenamiento, para este caso se realizó 500200 epochs.
- Tamaño de la imagen: aquí se hace referencia al tamaño de la imagen que procederá para la red. El mismo que será fijado en la programación ya que encajaría con el resto de la red y este se lo colocará para que la imagen tenga una resolución de 416 píxeles. Este tamaño se lo adecua al momento de

ejecutar el entrenamiento puesto que a mayor imagen mayor tiempo de entrenamiento y consumo de procesamiento.

- **Train_time:** este es el valor que se le asigna a la cantidad de veces a la cual se le va a poner a entrenar a una misma imagen.
- **Save_weth_name:** una vez culminado el proceso de entrenamiento de la red, todos los pesos de entrenamiento son guardados en el mismo archivo el cual se procederá a usar para realizar las predicciones al momento de ejecutar el script de reconocimiento.

3.8.4.5.2. Operaciones Principales de la red neuronal

Las operaciones principales. De la red neuronal si las describe a continuación.

A continuación, se tiene como entrada las imágenes adquiridas en los cultivos micro fans de lechuga. Un ejemplo de la foto que se tomó a continuación Liberación para crear el dataset se lo muestra en la fig 41.



Fig 41 imagen para el data set

Fuente: Autoría Propia

Al inicio, la red recibe píxeles de imagen como entrada. Los datos utilizados en este proyecto son imágenes de 960 x 720 píxeles, equivalente al uso de 691.200 neuronas. Pero sucede cuando se usa un solo color. Escala de grises En este caso la imagen es coloreada y

requiere 3 canales RGB (rojo, verde, azul), quedando $960 \times 720 \times 3 = 2073600$ neuronas. Estas neuronas forman la capa de entrada.

El preprocesamiento de las redes neuronales convolucionales se realiza internamente por lo cual los valores entre 0 y 1 deben convertirse antes de ingresar a la red. Entonces todo se divide por 255. El valor 255 se debe a que los valores de color de los píxeles están entre 0 y 255.

Después del preprocesamiento, la red neuronal convolucional realiza el procesamiento de discriminación. Es decir, se realiza la convolución. Consiste en tomar un grupo de píxeles cercanos de la imagen de entrada y manipular matemáticamente la salida escalar con una matriz (kernel). El tamaño del núcleo nos permite visualizar todas las neuronas de entrada de izquierda a derecha y de arriba a abajo. Esto produce una nueva matriz de salida que será una nueva capa de neuronas ocultas. La Figura 42 muestra la imagen de entrada y la Figura 43 muestra el tamaño del kernel para esta imagen.



Fig 42 Imagen de entrada

Fuente: Autoría Propia

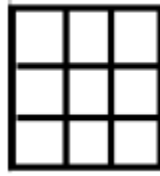


Fig 43 Kernel

Fuente: Autoría Propia

El kernel toma valores aleatorios para seguir una distribución normal. Luego están los filtros (conjuntos de núcleos). En este caso, no se aplica un solo kernel, sino muchos kernels, llamados filtros. En este caso hay 256 filtros como se muestra en la Figura 44. Hay 256 matrices de salida llamadas mapeos de características.

```
73 [convolutional]
74 batch_normalize=1
75 filters=256
76 size=3
77 stride=1
78 pad=1
79 activation=leaky
```

Fig 44 Filtros para la primera capa oculta de las neuronas

Fuente: Autoría Propia

De las 256x256x1 cada uno para un total de 65536 neuronas en la primera capa oculta de neuronas. La figura 45 muestra un kernel de ejemplo que realiza un producto de matriz en la imagen de entrada y mueve un píxel de izquierda a derecha y de arriba a abajo. Esto produce una nueva matriz que forma un mapa de características.

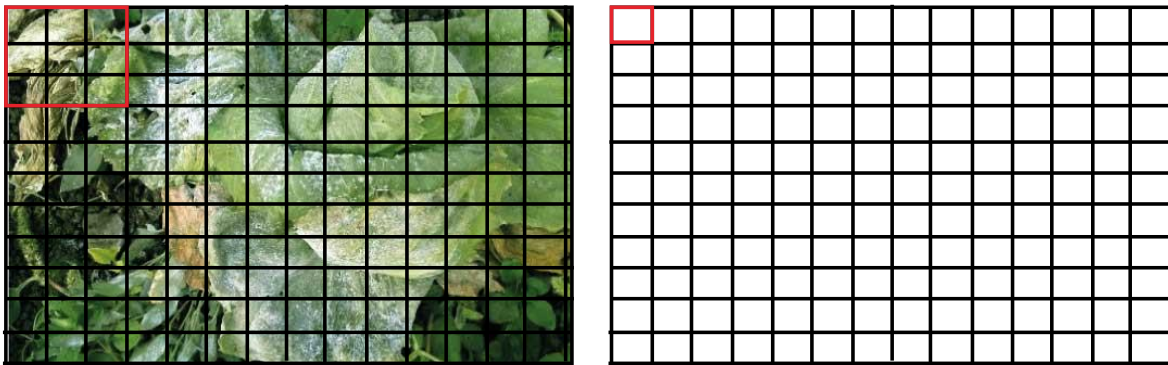


Fig 45 Producto Matricial con la imagen de entrada del kernel

Fuente: Autoría Propia

A medida que el kernel se mueve, se recopilan nuevas imágenes filtradas del kernel. Es como tener 256 nuevas imágenes filtradas como se ve en la primera convolución. Extrae características específicas de la imagen original para ayudar a distinguir más adelante un objeto de otro. En esta red neuronal convolucional se utilizó la función de activación ReLu (Unidad Lineal Rectificadora). Para continuar con las operaciones que se realizan en las redes neuronales convolucionales, tenemos el muestreo o submuestreo⁸. En este paso, se muestrean las neuronas más representativas antes de la reconvolución. El muestreo se realiza para reducir el tamaño de la siguiente capa de neuronas mientras se conservan las características clave reconocidas por cada filtro. Hay varios tipos de muestreo. Se utilizó la agrupación máxima. Utilice el método de agrupación máxima con un tamaño de 2 por 2, como se muestra en la Figura 47

1 (tamaño asignado en la configuración de red). Esto significa que solo se toma el tamaño asignado y se mantiene el valor máximo de

píxeles. Es decir, la imagen se reduce a la mitad y se mantiene en 128 x 128 píxeles. Esta técnica conserva la información más importante para reconocer la función deseada.

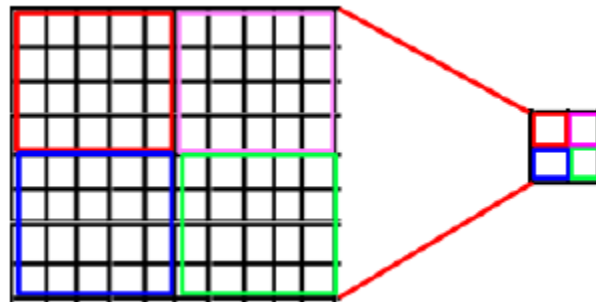


Fig 46 Técnicas de Max-Polling 2x2

Fuente: Autoría Propia

Una vez que se aplica la técnica de agrupación máxima, se realiza la primera convolución. La figura 48 muestra la representación de la primera convolución. Consta de una entrada y una serie de filtros para generar mapas de características y submuestreo. Dado que la imagen está en color, piense en el resultado como si tuviera 3 niveles.

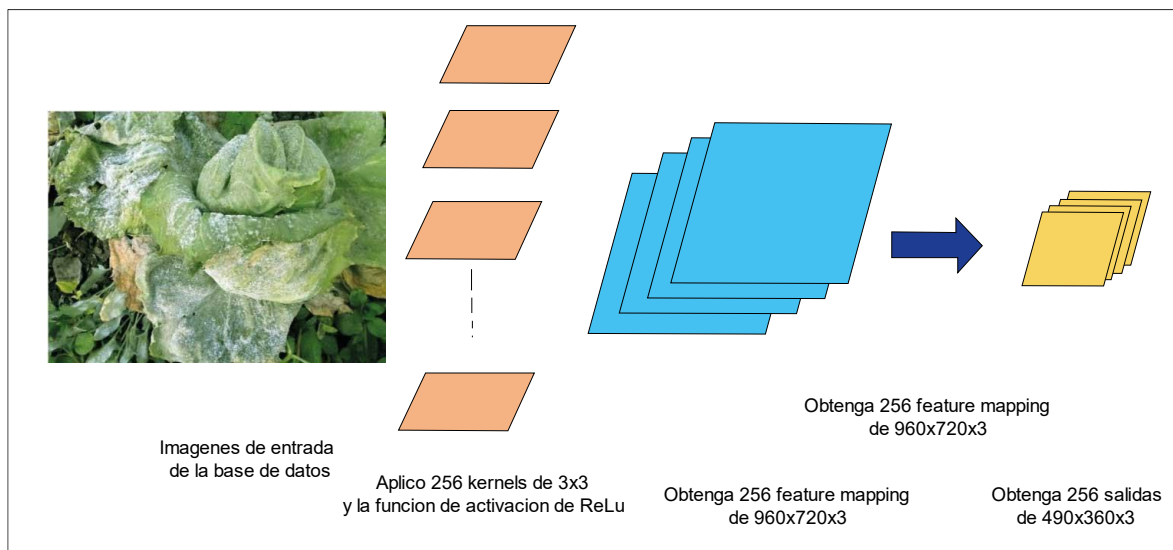


Fig 47 Primera Convolución

Fuente: Autoría Propia

La primera convolución captura las características básicas, es decir, líneas o curvas. A medida que la convolución crea más capas, el mapa de características se vuelve capaz de reconocer formas más complejas, lo que produce el reconocimiento deseado.

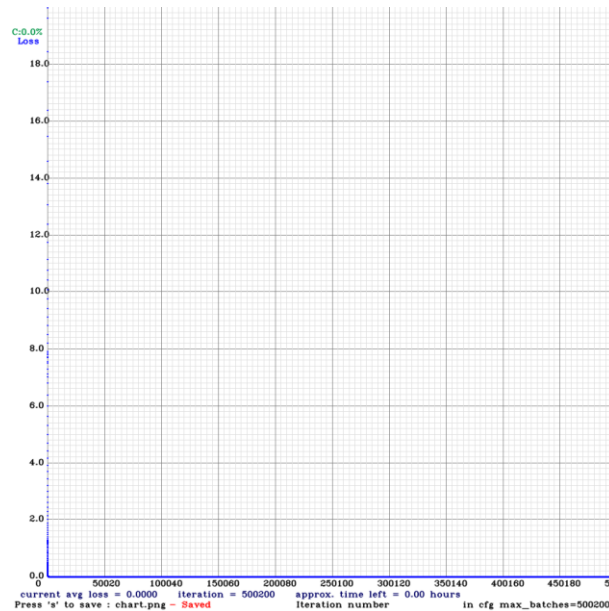


Fig 48 Mapa de Convoluciones

Fuente: Autoría Propia

Como se puede observar, se realizan algunas convoluciones, en donde se siguen los pasos anteriormente realizados, hasta llegar a la última convolución sin tener pérdida de currents así como una interacción de 500200 batches.

3.8.4.6. Reconocimiento de *Erysiphe cichoracerum* en la Lechuga

Una vez que el modelo construido en Yolo Tiny V3 se entrena en la web oscura, podemos ejecutar pruebas para detectar *Erysiphe cichoracerum* en cultivos Micro-Farms de lechugas, como se muestra en la Figura 49 donde se demuestra la detección con un cuadro delimitador de color Azul.

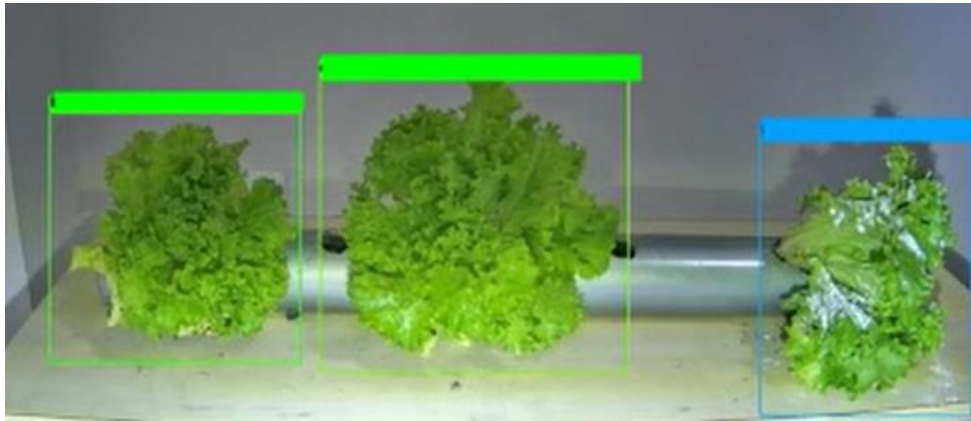


Fig 49 Erysiphe reconocida con el modelo entrenado

Fuente: Autoría Propia

La Figura a continuación muestra cómo el modelo entrenado detecta la enfermedad en tiempo real. Una Enfermedad detectada se marca con un cuadro delimitador. Tenga en cuenta que la darknet oscura permite la detección en tiempo real. Para entrenar la red convolucional.

contamos con Yolo Tiny V3 con interfaz Darknet. El archivo custom.cfg es un archivo de configuración en el que se establecen parámetros como capas, clases y filtros de redes neuronales según las necesidades del usuario. Después de completar el entrenamiento con el archivo de pesos generado, se ejecuta cada prueba de reconocimiento de objetos.

4. PRUEBAS DE FUNCIONAMIENTO.

En este capítulo se realizará las pruebas respectivas del sistema de aprendizaje de máquinas en cuanto al reconocimiento de la enfermedad de las lechugas *Erysiphe cichoracearum*, realizado en el capítulo 3. Por lo cual se procedió a utilizar una cámara para el uso de adquisición de datos, las cuales no deben tener una buena resolución. De tal manera que se usó una cámara profesional utilizada como cámara web; así como la consideración del objeto el cual se lo va a reconocer (*Erysiphe cichoracearum*) para proceder con el proceso de pruebas de laboratorio, mediante la adquisición de datos de la enfermedad de la lechuga para su respectivo reconocimiento y posterior entregar los resultados y conclusiones obtenidas.

4.1. Adquisición de datos

Al empezar a trabajar con Redes Neuronales Convolucionales o Visión por Computador la recolección de datos y el uso de estos son de mucha importancia, ya que, mientras recolectes más datos al sistema se lo puede hacer mucho más eficiente; así mismo como la base de datos y tan robusta al momento de procesar el entrenamiento vamos a ocupar mucho más RAM y GPU.

Al trabajar directamente con el procesador del computador vamos a tener menos procesamiento por las limitaciones del equipo; sin embargo, se las solventa usando la GPU adicionales de la tarjeta gráfica nvidia (nvidia gforce 1650ti), así como también el procesamiento del sistema de Microsoft Neural Stik 2 quien adicional más procesamiento al computador haciendo que este no tenga problemas al momento de realizar el entrenamiento de todo el algoritmo.

Para el proceso de toma de información en este caso de fotografías para nuestra base de datos se realizan varias pruebas, en las cuales tenemos resultados erróneos y favorables dependiendo de la resolución en la cual vamos teniendo las fotografías por lo cual se puede llegar a tener más eficiencia en el entrenamiento que tienen una resolución de 960*720 pixeles como se muestra a continuación en la Fig 50.

A principio del proyecto se tomaron fotografías de alta resolución las cuales hicieron que a la hora de las pruebas en el entrenamiento se demore mucho más de lo habitual, por lo que se redujo el tamaño de la imagen y se mejoró así el tiempo de entrenamiento y asertividad en la ejecución del script el cual tuvo un entrenamiento de 925 imágenes con 500200 interacciones o batches en 120 horas



Fig 50 Cultivo Micro-Farm Diferentes Etapas de crecimiento

Fuente: Autoría Propia



Fig 51 Cultivo Micro-Farm 3 Niveles en Vertical

Fuente: Autoría Propia

4.2. Pruebas del Sistema

Por parte de las pruebas a realizar, previamente se debe tener a consideración si el sistema se lo va a implementar en un lugar fijo o si se va a realizar pruebas en movimiento o pruebas a escala de laboratorio; por lo tanto, se llega a obtener una estación de trabajo para el sistema la cual muestra la Fig 52 a continuación donde se tiene una cámara full HD así como la iluminación de un reflector para evitar la falta de luminiscencia para la cámara, así como en la Fig 53 se puede evidenciar el montaje de la cámara frente al cultivo en vertical Micro-Farm para que funcione en conjunto con el algoritmo de reconocimiento.



Fig 52 Sistema de Reconocimiento De Imagen

Fuente: Autoría Propia



Fig 53 Implementación del Sistema Completo con el Cultivo en Vertical

Fuente: Autoría Propia

Por lo cual, una vez instalado el sistema y se encuentre en operatividad tanto la cámara como la lámpara que ayudara con la luminiscencia para no perder detalle alguno de las imágenes, se empieza a realizar pruebas del funcionamiento con toma de videos e imágenes en tiempo real, mientras tanto a continuación se detalla de la misma manera el diagrama de conexión que se realizó para que el funcionamiento de nuestro Algoritmo de Reconocimiento de imágenes Funcione correctamente u no nos arroje falsos positivos en los cultivos.

4.2.1. Diagrama de Conexión basados en la arquitectura del sistema

En esta sección se integran todos los elementos de todo el sistema. Por ello, se considera que en el caso del esquema grafico que se muestra a continuación en la Fig 54. A través del sistema computacional y la conexión a través del puerto del computador con la cámara, así como el puerto de conexión al monitor exterior donde se visualizara la detección de *Erysiphe cichoracerum*.

ARQUITECTURA



Fig 54 Arquitectura el Sistema

Fuente: Autoría Propia

Una vez que se programa nuestro script para que este ingrese al sistema y se lo inicialice automáticamente cada dos semanas durante 2 minutos este comenzara con el reconocimiento, se ejecutara el script ubicado en la carpeta de Darknet; donde se encuentra la subcarpeta llamada training en la cual se encuentran los archivos completos del sistema en la carpeta llamada model_lech.data, como se puede apreciar en la imagen Fig 55.

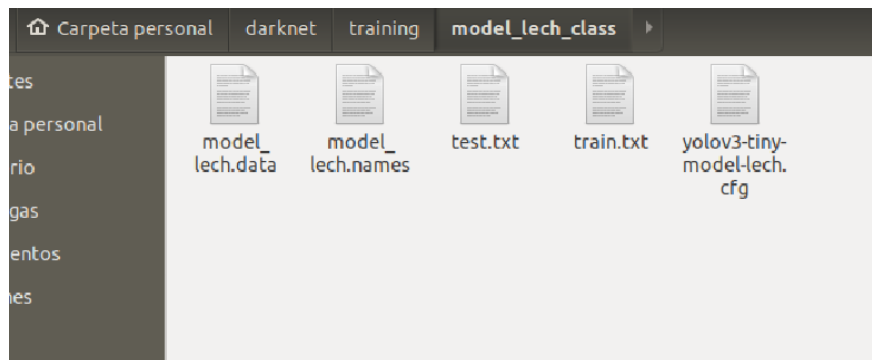


Fig 55 Archivos de Ejecución de Todo el Sistema

Fuente: Autoría Propia

Al considerar las pruebas que se van a realizar son a escala de laboratorio, del cual se pone a consideración que las mismas pruebas se las puede realizar a través de:

- Mediante imágenes Guardadas
- Mediante videos
- En tiempo real

A continuación, se detallan las pruebas que se procedieron a realizar en tiempo real

4.2.2. Pruebas del sistema en tiempo real

Para empezar con las pruebas de todo el sistema en tiempo real, se procede a ejecutar el código mencionado en el anexo con el cual se procede con la ejecución del mismo y la visualización del algoritmo en tiempo real en la detección de Erysiphe cichoracerum, la cuales se muestran los resultados continuación en la Fig 56. Donde se muestra como es el funcionamiento de este mostrando los cuadros de limitación, así como el porcentaje de precisión y etiqueta dada con anterioridad.

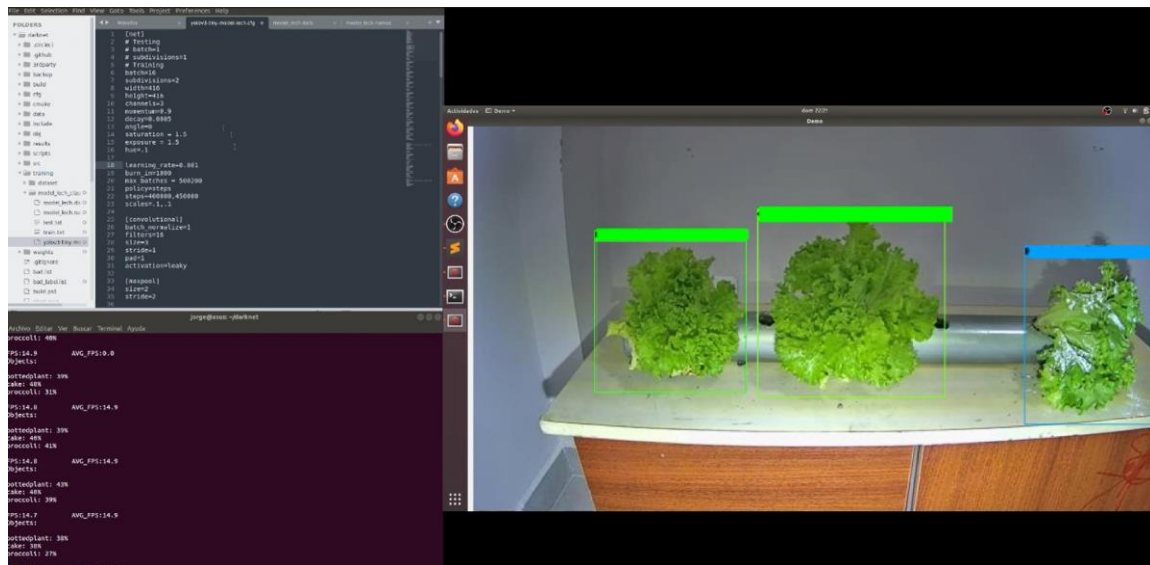


Fig 56 Funcionalidad en tiempo Real del Algoritmo EN base al Entrenamiento de Lechugas

Fuente: Autoría Propia

En la Fig 57, se observa claramente el resultado final del reconocimiento de la enfermedad Erysiphe cichoracearum, con lo cual se muestra la eficiencia en cuanto a la precisión que se tiene con el algoritmo de aprendizaje de maquina con cuadros delimitadores que permiten saber exactamente cuál es la lechuga afectada.

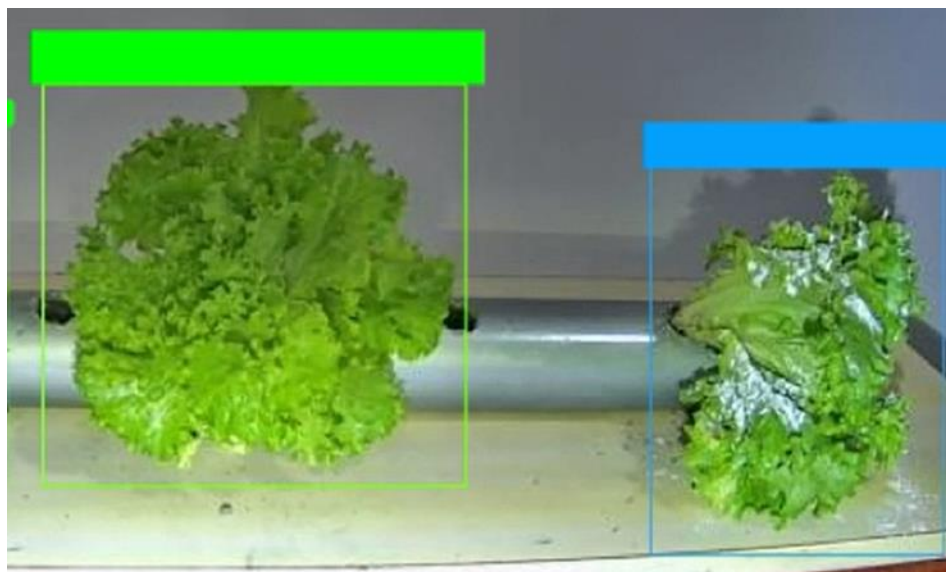


Fig 57 Cuadro Delimitador de lechuga Sana o Enferma

Fuente: Autoría Propia

Se toman otras muestras para verificar la precisión cuando se ejecuta el sistema en tiempo real en la Fig 58, se verifican así mismo la precisión en canto a la enfermedad detectada así mismo se muestra los resultados cuando se tiene a varias lechugas juntas, de la misma manera el algoritmo las empezara a detectar como sanas si es el caso de que no tengan la enfermedad del *Erysiphe cichoracearum*

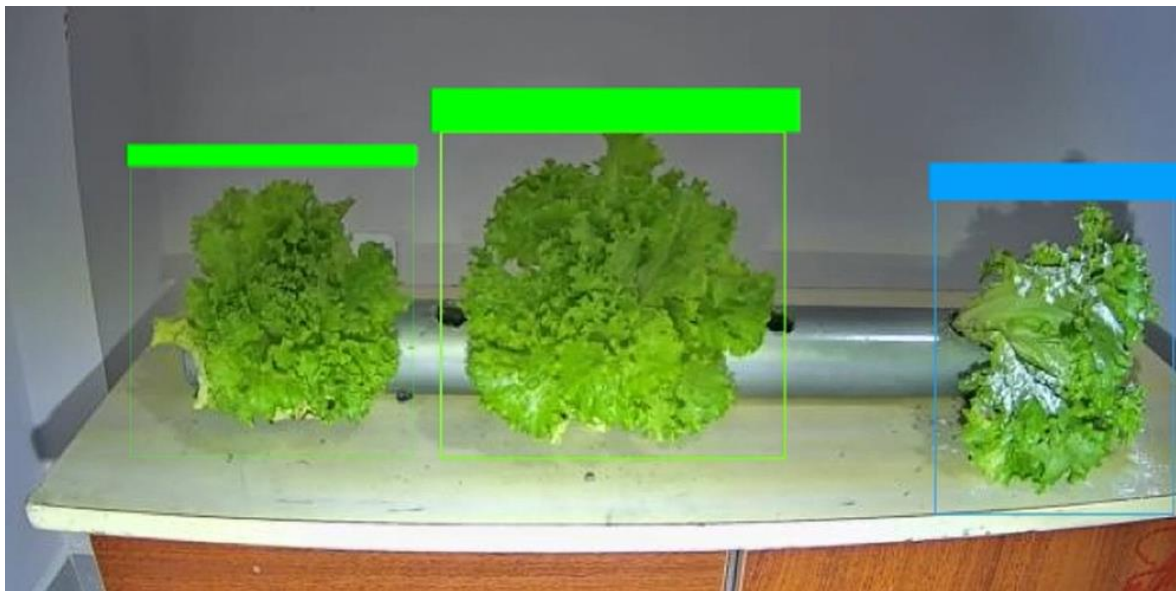


Fig 58 Precisión el Reconociendo en Tiempo Real

Fuente: Autoría Propia

Como puede ver en la siguiente figura, cuando se ejecuta automáticamente, el programa se ejecutará a la misma hora a las 9:00 a. m. en mejores condiciones de luz, lo que hace que la detección de enfermedades sea óptima y efectiva. De igual forma, se realiza cada dos semanas ya que da tiempo suficiente para ver el crecimiento de la lechuga y el impacto en el cultivo.



Fig 59 Niveles de Cultivos Micro-Farm de Lechugas

Fuente: Autoría Propia

Aquí puede ver cómo este sistema ya está funcionando a pequeña escala de cosecha Micro-Farm. Queda claro que no solo detecta la enfermedad de *Erysiphe sicolaceram* de forma lineal, sino también en diferentes niveles y posiciones de la lechuga enferma; de la misma forma el sistema estaría realizando el reconocimiento de imagen en tiempo real en varias etapas del crecimiento de las lechugas, así como en diferentes niveles de cultivos como se lo puede apreciar en la Fig 61 en donde tenemos el cultivo de lechugas en vertical y en un orden específico; por otro lado tenemos en la Fig 62 el mismo sistema fusionando con la diferencia que fuimos cambiando de orden las lechugas para poder demostrar que el sistema funciona de la manera más adecuada y detecta la enfermedad en todo el cultivo sin importar en qué posición se encuentre la lechuga afectada.



Fig 60 Funcionamiento Cultivo Enfermo Sano

Fuente: Autoría Propia



Fig 61 Funcionamiento del cultivo Enfermo y Sano

Fuente: Autoría Propia

De manera similar, al evaluar datos que se muestran directamente en un monitor, se podrá tener a la vez un video de respaldo de la ejecución del nuestro algoritmo de reconocimiento de imágenes guardado como un archivo en MP4 en una carpeta alojada en el directorio del computador, al momento en que se termine de realizar la ejecución del sistema y la evaluación de las lechugas en tiempo real. De esta forma, el medio en el cual se están criando las lechugas estaría más controlado,

4.2.3. Medidas Preventivas para Cultivos Enfermos

Para tener a consideración en este apartado se presentan los medios preventivos para que los cultivos puedan desarrollarse más sanos y no tener perdidas de los cultivos, así de esta manera y si se notase que el cultivo se encontrara comprometido se tendrían que tomar medidas preventivas como las que se menciona a continuación.

- El uso de cal para tratar de eliminar y evitar que la enfermedad siga propagándose por todo el cultivo (según lo mencionado en la entrevista de los cultivos tradicionales)
- A considerar los fungidas para eliminar por completo el hongo en la lechuga
- Septum, una poderosa solución natural de acción fungicida donde las moléculas clave procedentes del extracto de Equisetum arvense
- Descarte completo de la lechuga comprometida, si es el caso de no tener consideraciones de cura.
- Tomar medidas de los nutrientes del agua y si es mayor de lo normal, realizar un reemplazo del agua para evitar que se siga enfermando el cultivo
- Tener a consideración colocar un medidor de humedad en el cultivo, y llevar un control de este.

Teniendo consideración a lo mencionado con anterioridad, la toma de decisiones en al momento de detectar un cultivo comprometido a través de nuestro algoritmo de reconocimiento de imagen en tiempo real, sería una prioridad esencial la revisión de los

nutrientes del agua o la temperatura de los cultivos. Porque estos son los factores principales causantes de esta enfermedad afectando directamente a los cultivos Micro-Farm de lechuga.

De la misma manera siempre es aconsejable tener en cuenta que cuando se tiene cultivos enfermos es apropiado sembrarlos o cultivarlos en ambientes con la suficiente ventilación así como la separación adecuada para que estos puedan crecer de la mejor manera, así como siempre optar por tratar de prevenir este tipo de enfermedades de la manera mas natural a usar fungicidas que son más dañinos para los cultivos e incluso después para el consumo.

4.2.4. Visión por computador y aprendizaje de maquina

La visión por computadora juega un papel clave en el monitoreo de la lechuga para la detección temprana de enfermedades (*Erysiphe cichoracerum*) a través de habilidades perceptivas que pueden identificar mejor el entorno de trabajo al realizar pruebas basadas en el reconocimiento de imágenes en tiempo real, que es para lo que principalmente fueron entrenadas; sin embargo, el aprendizaje automático, es menos viable para el proyecto actual porque este tipo de estrategia requiere que el sistema aprenda constantemente de la experiencia pasada y opere sin necesidad de programación, haciendo que no sea viable el uso de este por que requeriría de mayor costo de procesamiento para el reconocimiento causando que el sistema tenga respuestas mucho más lentas o a su vez haciendo que colapse siendo así una desventaja al muy importante en cuenta al sistema de visión por computador.

Basándose en el análisis anteriormente mencionado y con lo acatado con anterioridad en el capítulo II, para tener un mejor rendimiento en cuanto a procesamiento de imágenes y respuestas inmediatas a la detección temprana de la enfermedad *Erysiphe Cichoracerum* mejor conocida como la enfermedad del oído, es mucho más recomendable usar los algoritmos de visión por computador ya que estos no requieren de una programación especifica en cuanto a la funcionalidad que se le va a dar que en este caso son detección de enfermedades en las lechugas de los cultivos Micro-Farms teniendo como resultado cultivos mucho más sanos que los convencionales.

4.2.5. Análisis de cultivos Tradicionales y Micro-Farm

Para tener una comparativa de por qué tener nuestro sistema de detección temprana de enfermedades ayuda a los cultivos Micro-Farms de lechuga a no tener pérdidas superiores al 3% y hacer que al momento de cultivar las lechugas estas sean mucho más grandes y sanas, se realizó una entrevista al dueño de los cultivos convencionales de lechugas el Sr. Juan Vaca quien lleva cultivando lechugas por ya más de 40 años en sus terrenos ubicados en el sector de Imbaya como se muestra en la Fig 62. a continuación.



Fig 62 Cultivos de Lechuga de la Finca San Rafael ubicada en el sector de Imbaya

Fuente: Autoría Propia

El señor Juan Vaca como se puede observar en la Fig. 64 es propietario de una gran cantidad de cultivos de lechugas de los cuales nos menciona que al momento de mantener un control en cuanto a plagas y en este caso enfermedades de la lechuga como lo es la del oído es mucho más difícil identificar cual es el cultivo afectado ya que no se tiene un control de las etapas de crecimiento, de la misma manera cuando se realiza una revisión del crecimiento y se identifica esta enfermedad se la trata de controlar con cal haciendo que muera este hongo o enfermedad de *Erysiphe cichoracearum*, al momento de la cosecha se identifica que los cultivos se ven afectados en un número considerable de lechugas las cuales para el llegarían a representar una pérdida del 26% del cultivo por cosecha e incluso se puede decir que en

ciertas circunstancias de excesos de lluvias se han visto mucho más afectados que han tenido perdidas del 50%.



Fig 63 Cultivos de Lechugas etapa de Cosecha revisión de cultivos Enfermos

Fuente: Autoría Propia



Fig 64 Plantaciones de Lechugas en Diferentes Etapas de Crecimiento

Fuente: Autoría Propia

Por lo tanto, se puede decir que, si contamos con un sistema de control temprano de este tipo de enfermedades, la calidad de los cultivos de las Micro-Farms será mejor y no se tendrían este tipo de pérdidas ya que se estaría tomando las debidas correcciones al momento de que se detecte que el cultivo se encuentra afectado, separando en este caso a la lechuga afectada para no dar tiempo a que esta enfermedad comience a afectar al resto del cultivo o tratando de curar como se menciona en el apartado anterior en las medidas preventivas, teniendo como referencia un mejor cultivo mucho más sano y de mejor calidad en cuanto al cultivo convencional como muestra la tabla comparativa a continuación.

Tabla 21 Comparativas Relevantes de los Cultivos Micro-Farm y Tradicionales

Cultivo Micro-Farm	Cultivo Tradicional
Cultivo mucho más sano en habientes controlados, ya que se tiene incluso una cantidad variada de cultivos ya que estos no están compitiendo por los nutrientes del suelo	No se tiene un control en cuanto al espacio y no sería adecuado para tener diversidad de cultivos porque no tendría nutrientes adecuados para cada uno de ellos
En este tipo de cultivos de lo puede realizar en cualquier lugar que no sea necesariamente en plano teniendo en cuenta que son mucho más prácticos y pequeños para el hogar	El suelo es una limitante ya que para este tipo de cultivos siempre vamos a requerir tierra y esta tiene que ser tratada para el tipo de cultivo que se vaya a tener
Garantiza una cosecha mucho más sana y saludable sin pérdidas exuberantes de cultivos en la cosecha debido a que están en un ambiente controlado	En este tipo de cultivos no se sabría exactamente cuándo se necesitaría aportes nutricionales el cultivo para que este crezca de mejor manera causando pérdidas
Son cultivos eco amigables con el medio ambiente ya que no necesitarían de ningún tipo de químico o pesticidas	Como se encuentran cultivadas en tierra son más propensas a enfermedades y plagas teniendo que fumigar cultivos completos para evitar plagas o enfermedades
Reduce tiempo y dinero ya que los cultivos al estar en un ambiente más controlado crían de	El tiempo de cosecha en este tipo de cultivos si tenemos como referencia la lechuga se demora 3

mejor manera y mucho más rápido en este caso la lechuga se demoraría 6 semanas en cosecharse	meses y medio aproximadamente desde su germinación
Los cultivos ahorran agua ya que utiliza un sistema de recirculación de agua aprovechando al máximo el riego del cultivo y la absorción de nutrientes teniendo un ahorro considerable del agua	Los cultivos tradicionales desperdician más agua ya que esta filtra al suelo directamente haciendo que se necesite mayor cantidad de agua para el riego
Mucho más sustentable y de pocos requerimientos haciendo que sea mucho más económico su mantenimiento y crianza dando como resultados lechugas mucho más sanas para el consumo	Tiene un costo mucho más alto en cuanto al cultivo y en etapas de cosecha se tiene pérdidas en plantas debido a que se presentan enfermedades que no son controladas

Considerando la Tabla 21 se demuestra que existen muchos más beneficios en cuanto a nuestros cultivos Micro-Farms, demostrado que se puede tener cultivos mucho más sanos e incluso más grandes que los cultivos tradicionales ya que están en ambientes más controlados, teniendo incluso beneficios sustentables diseñando un sistema Micro-Farm mucho más controlado y automatizado, incluso se puede obtener comercialización de los mismos teniendo beneficios económicos al obtener lechugas mucho más grandes y saludables.

CONCLUSIONES Y RECOMENDACIONES.

Conclusiones

Partiendo de la base teórica de detección enfermedades de la lechuga mediante visión artificial y a través de redes neuronales, se concluye que existen varios tipos de redes neuronales a utilizar, de las cuales se utiliza la más adecuada. Utilizándose así una red neuronal convolucional para la implementación del sistema actual.

Como se mencionó en la sección de marco teórico, existen diversos tipos de enfermedades y virus. Por lo cual, se utilizó la propia especie de lechugas para la detección. Con esta información sobre las enfermedades, se pudo concluir que para este sistema se seleccionó una enfermedad por las condiciones en las que se forman, ya que eran óptimas en su formación, a diferencia de las demás que se dan en un cultivo diferente.

Se ha validado un entrenamiento con redes neuronales convolucionales con el propósito de extraer características profundas de las imágenes. Esta vez, extrajimos las características de una enfermedad que se da en cultivos Miro-Farm. Estas características tienen como prioridad mostrarse como manchas blancas en la hoja de la lechuga, así como varios patrones de código de la enfermedad. A partir de esto, podemos concluir que la adquisición de un número considerable de datos es un factor clave para la precisión del modelo de las CNN, ya que las redes neuronales convolucionales tienen más éxito cuando se las entrenan con una gran cantidad de datos.

Se llegó a la conclusión de que al momento de ejecutar un sistema de aprendizaje de máquina directamente del computador en conjunto a un sistema de aceleración como lo es el Neural Stik de Microsoft es importante tener en consideración la compatibilidad del uno con el otro ya que este será una ayuda importante al momento del uso de la GPU del computador teniendo resultados en menos tiempo.

Debido a que, en una red neuronal se tiene dos fases: una fase en la cual se dedica solo al entrenamiento y la segunda fase es para su validación; se concluye que, en las dos fases se utiliza datos completamente distintos por lo que se realiza la toma de datos, para el entrenamiento en el que se agrupa imágenes para la extracción de características esenciales y para su validación que es en donde se necesita imágenes para comprobar la red previamente entrenada.

Con la utilización de un Neural Stick, se pudo concluir que, el mismo permitió ejecutar y programar en Python directo desde nuestro pc, ya que dispone de la ventaja de brindar al computador la GPU necesaria, lo que brindó mayor facilidad para realizar el entrenamiento.

Se pudo comprobar que el Tiny Yolo V3 puede ser entrenado y dar resultados óptimos y precisos en cuanto al reconocimiento de imágenes; por otro lado, las pruebas realizadas se las puede hacer directamente en tiempo real, gracias a la precisión de Tiny Yolo V3 ya que es hasta 5 veces más rápido que el normal de Yolo V3 lo cual no baja el rendimiento del sistema computacional, siendo este que fue entrenado basándose en lechugas con una gran Data Base las cuales se entrenaron durante 120 horas con una interacción de 500200 batches.

Al ejecutar todo el sistema probar y validar los resultados en tiempo real de detección de la enfermedad de Erysiphe cichoracearum, podemos decir que se ha desarrollado un sistema de visión por computador que puede detectar objetivos específicos como se esperaba.

Teniendo todo el sistema desarrollado, se tienen pruebas reales del funcionamiento del mismo y así demostramos que se tienen cultivos muchos mas grandes y sanos teniendo beneficios sustentables con un diseño Micro-Farm convencional e incluso beneficios económicos

Recomendaciones

Para ajustar los pesos y optimizar el modelo, se recomienda observar más de cerca la estructura y el comportamiento de la red neuronal convolucional en cada capa.

Para mejorar la eficiencia del sistema, se recomienda aumentar la memoria RAM y el rendimiento del procesador al implementar modelos de entrenamiento, la cual se solventaría con el acelerador Neural Stick 2

Es deseable disponer de una gran cantidad de datos para entrenar la red neuronal de tal forma que el aprendizaje sea más preciso y se equilibren las características de las muestras y sus pesos.

Por otro lado, el uso de la herramienta LabelImg es muy importante durante la etapa de etiquetado. Al ser en formato YOLO compatible con la versión Tiny Yolo V3, se genera un archivo (.txt) con información de cada imagen en forma de coordenadas siendo estas utilizadas para entrenar el sistema de reconocimiento.

Tiny Yolo V3 se puede entrenar y se ha demostrado que proporciona los mejores resultados en lo que respecta a la detección de objetivos específicos. Por otro lado, las pruebas que se ejecutan se las realiza en tiempo real, por lo que no afectan el rendimiento del sistema gracias al acelerador de GPU.

A través de las pruebas de rendimiento, pudimos confirmar el uso exitoso del sistema de redes neuronales convolucionales utilizando la estructura de Yolo Tiny V3, que ha demostrado ser muy preciso.

Al realizar pruebas y verificar los resultados de la detección de enfermedades (Erysiphe cichoracearum), podemos decir que hemos desarrollado un sistema óptimo y eficaz que puede detectar objetivos específicos como se lo esperaba.

5. ANEXOS

5.1. Estructura Principal de la Programación del Sistema

Para la ejecución del sistema completo primero se tiene que instalar los requerimientos necesarios del sistema los cuales se detallan continuación en la Figura 66. Así como también los archivos y comandos principales que se deben ejecutar al momento del entrenamiento de archivos weight y la ejecución del sistema ya entrenado

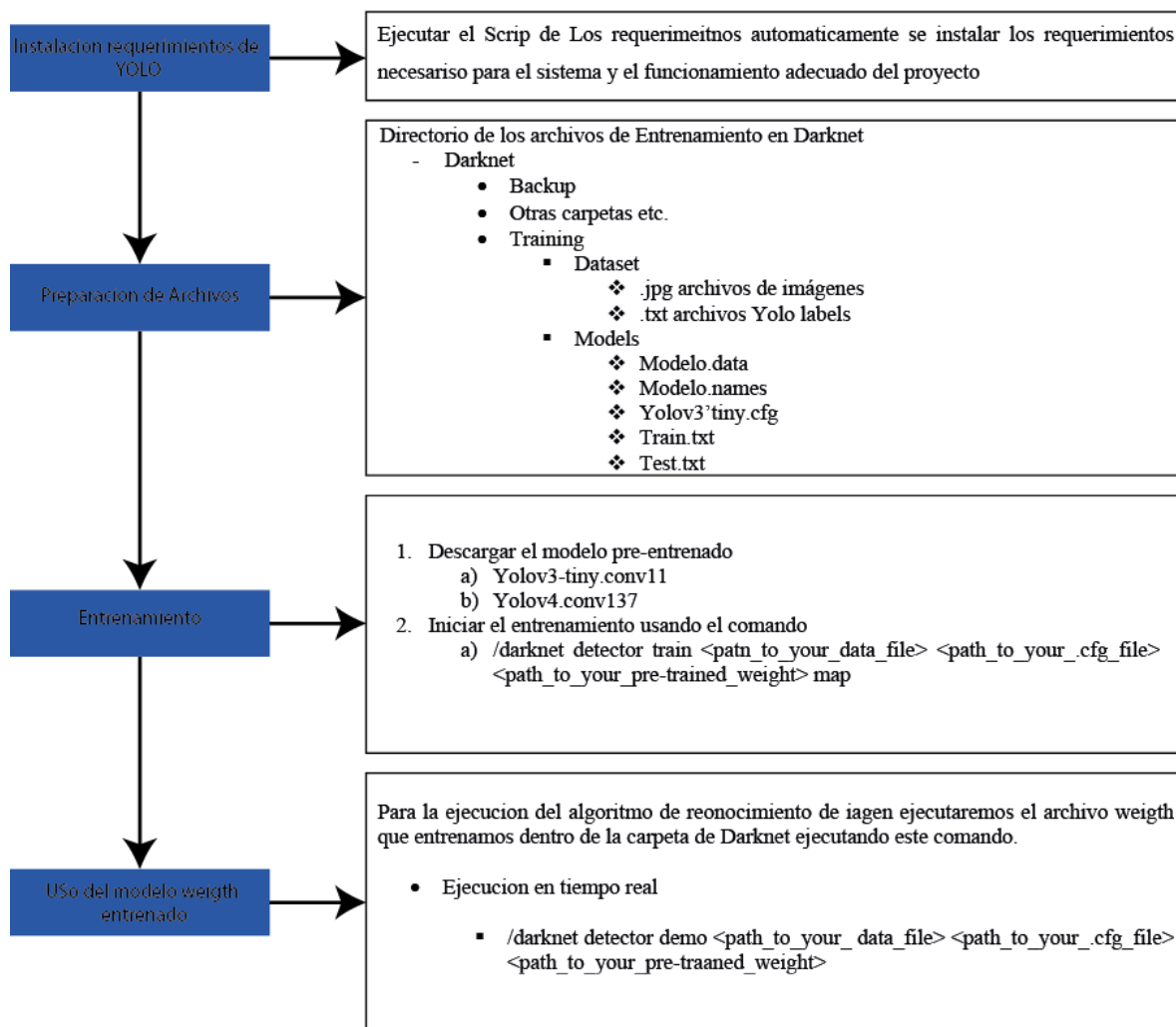


Fig 65 Estructura de Instalación y Creación de archivos para la Ejecución del Sistema

Fuente: Autoría Propia

5.2. Rutas y comandos Esenciales en la Programación

En el modelo de para el entrenamiento de nuestro algoritmo se tiene que direccionar una ruta específica en la cual se van a tener los archivos principales para el entrenamiento del archivo el cual se va a proceder ejecutar en cuanto previo el entrenamiento, este archivo contendrá estas rutas, así como también el número de clases o en este caso el número de variables que se van a reconocer como se muestra a continuación en la Figura 67.

```
classes= 2
train  = /home/jorge/darknet/training/model_lech_class/train.txt
valid  = /home/jorge/darknet/training/model_lech_class/test.txt
names  = /home/jorge/darknet/training/model_lech_class/model_lech.names
backup = /home/jorge/darknet/backup/
```

Fig 66 Rutas de los archivos para el pre-entrenamiento del algoritmo

Fuente: Autoría Propia

Archivo de ejecución Makefile aquí primordialmente se tiene las variables principales para ejecutar correctamente el sistema, en las cuales como se puede apreciar en la Figura 68 existen las variables de la GPU, CUDNN entre otras estas se tendrán que activar mediante unos y ceros los cuales indican el estado de activo o inactivo, en este caso se tendrán que activar las que se muestra a continuación para tener un buen uso de toda la GPU del computador, así como las del Neural Stick II de Microsoft.

```
GPU=1
CUDNN=1
CUDNN_HALF=0
OPENCV=1
AVX=1
OPENMP=1
LIBSO=0
ZED_CAMERA=0
ZED_CAMERA_v2_8=0
```

Fig 67 Archivo de Ejecución Previo Activación de GPU y aceleradores Gráficos

Fuente: Autoría Propia

6. BIBLIOGRAFIA

- Abril Cañas, M., Valencia, M. L., JOHANA RESTREPO Ingeniero Electricista Profesor Catedrático, B., ANDRÉS HOLGUÍN Ingeniero Electricista, G., & Asistente, P. (2007). Computer Vision System for Real-time Registry of Pedestrian Density. *Scientia et Technica Año XIII*, 35.
- Acuña, A. (2019). Visi n artificial aplicada a la detecc i n e identificaci n de personas en tiempo real. *Escuela {Polit} Cnica {Nacional}. {Agrocalidad}. (2015). {Gu} a {De} {Buenas} {Pr} Cticas {Agr} Colas {Para} {Tomate} {Ri} n. 80. [Http://Www.Agrocalidad.Gob.Ec/Wp-Content/Uploads/2016/07/Guia-Tomate-Rinon-Final.Pdf](http://Www.Agrocalidad.Gob.Ec/Wp-Content/Uploads/2016/07/Guia-Tomate-Rinon-Final.Pdf) {Aguirre}, {H}. (2019). {DESARROLLO} {Y}, 38(5), 25.*
- Aguirre, H. (2019). DESARROLLO Y {AN} LISIS {DE} {SISTEMAS} {DE} {ESTIMACI} N Y {DETECCI} N {DE} {OBJETIVOS} {DE} {RADAR} {MEDIANTE} {ALGORITMOS} {DE} {MACHINE} Y {DEEP} {LEARNING}. Universidad de las fuerzas armadas.
- Analogy, Everingham, M., Gool, L. Van, Williams, C. K., Winn, J., Zisserman, A., -Y., C., Gordon, D., Kembhavi, A., Rastegari, M., Redmon, J., Fox, D., Farhadi, A., He, K., Zhang, X., Ren, S., Sun, J., Huang, J., Rathod, V., ... Alemi, A. A. (2017). Lin. M. {Maire}, {S}. {Belongie}, {J}. {Hays}, {P}. {Perona}, {D}. {Ra}manan, {P}. {Dollar}, and {C}. {L}. {Zitnick}. {Microsoft} Coco: {Com}- Mon Objects in Context. {In} {European} Conference on Computer Vision, Pages 740 755. {Springer}, 2014, 3(3), 3316.
- Andrade Guevara, R. E. (2022). Sistema autónomo de control y supervisión de condiciones nutricionales del agua mediante una red de sensores y algoritmos de aprendizaje supervisado para el regadío en cultivo de espinaca bajo condiciones de hidroponía NFT vertical. <http://repositorio.utn.edu.ec/handle/123456789/12233>
- Antontio, J., & C., T. (2006). Sistema de visi n artificial para el reconocimiento y manipulaci n de objetos utilizando un brazo robot. Test, 1 125.
- BETANCOURT, G. A. (2005). LAS M QUINAS {DE} {SOPORTE} {VECTORIAL} ({SVMs}). 27, 67 72.
- Borovkov, A. (2017). Image Classification with Deep Learning. UNIVERSIT T {HAMBURG}.

- Botanica - Agricultura - Libro - Horticultura Herbacea Especial (Maroto Borrego JV - Mundi Prensa 1983) PDF | PDF.* (n.d.). Retrieved May 15, 2022, from <https://es.scribd.com/doc/260277972/Botanica-Agricultura-Libro-Horticultura-Herbacea-Especial-Maroto-Borrego-JV-Mundi-Prensa-1983-pdf>
- Caffe, O. (2019). *Intel® Neural Compute Stick 2 | Product Sheet.*
- Canziani, A., Paszke, A., & Culurciello, E. (2016). *An Analysis of Deep Neural Network Models for Practical Applications.*
- Carlos, J., & Pérez, M. (2018). *Aprendizaje supervisado para la detección de amenazas {Web}.*
- Cassidy, S., Coulter, M., Finkelston, T., Hoherchak, K., Mendes, A., Ott, G., Patton, C., Stein, K., Etienne, B., Louis, G., & Lerda, M. (2020). Hydroponic Crop Cultivation (HCC) for Food Security in Small Island Developing States. *2020 Systems and Information Engineering Design Symposium (SIEDS)*, 1–5. <https://doi.org/10.1109/SIEDS49339.2020.9106658>
- Cha, Y.-J., Choi, W., & Buyukozturk, O. (2017). Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. *Computer-Aided Civil and Infrastructure Engineering*, 32.
- Chacua Criollo, B. E. (2019). *Diseño de un sistema prototipo de reconocimiento facial para la identificación de personas en la Facultad de Ingeniería en Ciencias Aplicadas (FICA) de la Universidad Técnica del Norte utilizando técnicas de Inteligencia Artificial.* <http://repositorio.utn.edu.ec/handle/123456789/9572>
- Cicero, I. E. (2018). *Utilización de redes neuronales convolucionales para la detección de tipos de imágenes [INSTITUTO TECNOLÓGICO DE BUENOS AIRES ITBA].*
- Cong, J., & Xiao, B. (2014). *Minimizing in Convolutional Neural Networks.* International Conference on Artificial Neural Networks, 281–290.
- De-La-Hoz, E. J., De-La-Hoz, E. J., & Fontalvo, T. J. (2019). Metodología de {Aprendizaje} {Automático} para la {Clasificación} y {Predicción} de {Usuarios} en {Ambientes} {Virtuales} de {Educación}. *Información Tecnológica*, 30(1).
- E, R., & Briega, L. (2012). (2012). Deep {Learning} - {Libro} online de {IAAR}.

- Enyinna, C., Ijomah, W., Gachagan, A., & Marshall, S. (2018). *Activation Functions: Comparison of Trends in Practice and Research for Deep Learning*.
- García, A. B. S. (2018). Aplicación de {Aprendizaje} en la identificación de obstáculos en el trayecto de vehículos. In *Universidad Militar Nueva Granada*.
- Goyal, P., Pandey, S., & Jain, K. (2018). *Deep Learning for Natural Language Processing*.
- Guerrero, J. M. (2015). SISTEMA DE VISIÓN PARA AGRICULTURA DE PRECISIÓN: IDENTIFICACIÓN EN TIEMPO REAL DE LÍNEAS DE CULTIVO Y MALAS HIERBAS EN CAMPOS DE MAÍZ / VISION SYSTEM FOR PRECISION AGRICULTURE: REAL-TIME CROP ROWS AND WEEDS DETECTION IN MAIZE FIELDS. In *Unpublished*. <https://doi.org/10.13140/RG.2.1.3737.8720>
- Guidelines for Growing Microgreens LESSONS LEARNED FROM ECO CITY FARMS*. (n.d.).
- Hatcher, A., Jung, S., Keegan, H., Le, T., Quach, H., Ward, C., Weisberg, J., & Louis, G. (2019). Assessing the Viability of a Fold-Out Hydroponic Farm for Humanitarian Relief Efforts in Dominica. *2019 Systems and Information Engineering Design Symposium (SIEDS)*, 1–5. <https://doi.org/10.1109/SIEDS.2019.8735624>
- Huang, G., Li, T., Wu, F., Chen, D., Van Der Maaten, L., & Weinberger, K. Q. (2018). *Multi-Scale Dense Convolutional Networks for Efficient Prediction Dynamic Neural Networks View project Multi-Scale Dense Convolutional Networks for Efficient Prediction*.
- ISO/IEC/IEEE. (2018). *ISO/IEC/IEEE 29148:2018(E)*, 1–104. <https://doi.org/10.1109/IEEESTD.2018.8559686>
- Jones, T. (2017). *Arquitecturas de aprendizaje profundo*.
- José Núñez Sánchez-Agustino, F., Kanaan, S., Carles, I., & Royo, V. (2016). *Diseño de un sistema de reconocimiento automático de matrículas de vehículos mediante una red neuronal convolucional*.
- Khokhlov, I., Babaev, A., Litvinenko, V., & Gorbachev, R. (2020). *Tiny-YOLO object detection supplemented with geometrical data 4 th Ilya Ryakin*. <https://www.flir.com/products/blackfly-gige>
- Labeling images and text documents - Azure Machine Learning | Microsoft Learn*. (n.d.). Retrieved January 29, 2023, from <https://learn.microsoft.com/en-us/azure/machine-learning/how-to-label-data>

- Lactuca, (, El, E. N., Santa, B., De, F. E., Parroquia, L. A., En, A., Cantón, E. L., Cristian, A., & Toapanta, D. S. (2013). *"INTRODUCCIÓN DE CINCO VARIEDADES DE LECHUGA*.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(436.).
- Lecun, Y., Kavukcuoglu, K., & Farabet, C. (2010). Convolutional Networks and Applications in Vision. In *ISCAS 2010-2010 {IEEE} International Symposium on Circuits and Systems: Nano-Bio Circuit Fabrics and Systems*.
- Li, T., Ma, Y., & Endoh, T. (2020). A Systematic Study of Tiny YOLO3 Inference: Toward Compact Brainware Processor With Less Memory and Logic Gate. *IEEE Access*, 8, 142931–142955. <https://doi.org/10.1109/ACCESS.2020.3013934>
- Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8693 LNCS(PART 5), 740–755. https://doi.org/10.1007/978-3-319-10602-1_48
- Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., & Alsaadi, F. (2016). A survey of deep neural network architectures and their applications. *Neurocomputing*, 234.
- M., C., & Montilla, A. (2019). *Visi n artificial aplicada a la detecc i n e identificaci n de personas en tiempo real*. Escuela Polit cnica Nacional.
- Matich Dami. (2011). (2011). *Redes {Neuronales}: {Conceptos} {B} Sicos y {Aplicaciones}. {Historia}*, 55.
- Mckay, J., & Marshall, P. (2002, November). *Action Research: A Guide to Process and Procedure*.
- Merchan, F., Cornejo, A., F, S. G., & J. (2018). Uso de {Redes} {Neuronales} {Convolucionales} para el {Reconocimiento} {Autom} tico de {Im} genes de {Macroinvertebrados} para el {Biomonitoreo} {Participativo}. In *KnE Engineering (Vol.*
- Mohammed, M., Khan, M. B., & Bashie, E. B. M. (2016). Machine learning: {Algorithms} and applications. In *Machine Learning: Algorithms and Applications (Issue December)*.

- Moncayo Suárez, K. F. (2021). *Visión por computador para reconocimiento de malezas en cultivos de tomate riñón de invernadero, mediante redes neuronales*. <http://repositorio.utn.edu.ec/handle/123456789/11567>
- Morales, E., & Escalante, H. J. (2012). *Arboles de {Decisi} on {Contenido} {Algoritmo} {ID}3 {C} omo le hace y manejo de ruido {Ruido} y {Sobreajuste} y de {Modelos} de {Regresi} on*. 1 66.
- Muñoz, G., & Blanco, D. (2018). Aplicaciones de {T} cnicas de {Inteligencia} {Artificial} {Basadas} en {Aprendizaje} {Profundo} (deep learning) al {An} lisis y {Mejora} de la {Eficiencia} de {Procesos} {Industriales}. *February 2018*, 75.
- Neapolitan, R. E. (2018). *Neural Networks and Deep Learning*. Artificial Intelligence, 389 411.
- Nilsen, M. (2018). *Neural Networks and Deep Learning*. Determination Press.
- Núñez, M. J. (n.d.). *Tramas en Redes de Computadoras*.
- Oropeza, C. (2007). *Modelado y {Simulaci} n de un {Sistema} de {Detecci} n de {Intrusos} {Utilizando} {Redes} {Neuronales} {Recurrentes}*. 15 25.
- Patel, S., & Patel, A. (2018). Deep {Leaning} {Architectures} and its {Applications} {A} {Survey}. *International Journal of Computer Sciences and Engineering*, 6.
- Patterson, J., & Gibson, A. (2017). *Deep Learning: A Practitioner s Approach* (O Reilly).
- Pineda, P. (2018). *Dise o De Un Sistema De Reconocimiento Autom tico De Veh culos Mediante El Uso De Redes Neuronales Profundas (Dnn)*. <http://repositorio.utn.edu.ec/bitstream/123456789/9136/1/04> {RED} 220 {TRABAJO} {DE} {GRADO}.
- Plagas y Enfermedades más importantes de la Lechuga - NEVAL. Ensayos de Eficacia*. (n.d.). Retrieved May 15, 2022, from <https://www.ne-val.com/plagas-enfermedades-mas-importantes-lechuga/>
- Redmon, J. (2013). Darknet: {Open} source neural networks in c. *Http://Pjreddie.Com/Darknet/*, 3. <http://pjreddie.com/darknet/>
- Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *ArXiv*.
- Robots, M. (2009). *Sistemas De Visi n Artificial*.
- Roura, S. I., Pereyra, L., & del Valle, C. E. (2008). Phenylalanine ammonia lyase activity in fresh cut lettuce subjected to the combined action of heat mild shocks and chemical

- additives. *LWT - Food Science and Technology*, 41(5), 919–924.
<https://doi.org/10.1016/J.LWT.2007.06.005>
- Ruetsch, G., & Oster, B. (2008). *Getting Started with CUDA*.
- Ruiz Gómez, C. M. (2022). *Sistema autónomo de monitoreo y control de condiciones ambientales en cultivos verticales hortícolas urbanos mediante una red de sensores*.
<http://repositorio.utn.edu.ec/handle/123456789/12578>
- Serrano, J. F. V., Díaz, A. B. M., Calle, Á. S., & Sánchez-Marín, J. L. E. (2016). *Visión por computador*. 6ta. <http://www.visionporcomputador.es/libroVision/libro.html>
- Tomás-Barberdán, F. A., Loaiza-Velarde, J., & Saltveit, M. E. (2019). Early Changes in Wound-induced and Ethylene-induced Phenylpropanoid Metabolism in Lettuce. *HortScience*, 31(4), 599f–5599. <https://doi.org/10.21273/HORTSCI.31.4.599F>
- Tripp, D. (2005). Action research: a methodological introduction. *Educação e Pesquisa*, 31.
- Vizcaya, R. (2018). *DEEP {LEARNING} {PARA} {LA} {DETECCION} {DE} {PEATONES} Y {VEHICULOS} {SOBRE} {FPGA} {TESIS}*. UNIVERSIDAD {AUTÓNOMA} NOMA {DEL} {ESTADO} {DE} MEXICO.
- Wang, W., & Yang, Y. (2019). Development of convolutional neural network and its application in image classification: a survey. *Optical Engineering*, 1.
- Wu, J. (2017). *Introduction to convolutional neural networks*. National Key Lab for Novel Software Technology, 1 31.
- Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into {Imaging}*, 9(4).
- Zolezzi, M. V., Saavedra Del Ing Agrónomo, G. R., La Platina, I., La Platina Fabio Corradini Ing Agrónomo, I. S., Sc, M., La Platina Alejandro Antúnez Ing Agrónomo, I. B., La Platina Sofía Felmer Ing Agrónoma, I. E., Rayentué Patricia Estay Ing Agrónoma, I. P., La Platina Paulina Sepúlveda Ing Agrónoma, I. R., & Esquivel Ricardo Del Río Boletín INIA N° 0, C. (n.d.). *Coordinadores responsables*.