



**UNIVERSIDAD TÉCNICA DEL NORTE**

**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y**

**REDES DE COMUNICACIÓN**

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE**

**INGENIERÍA EN ELECTRÓNICA Y REDES DE COMUNICACIÓN**

**TEMA:**

**“SISTEMA EMBEBIDO PARA RECONOCIMIENTO Y CONTEO DE**

**ESPORAS DEL HONGO DE GÉNERO TRICHODERMA MEDIANTE**

**TÉCNICAS DE VISIÓN ARTIFICIAL”**

**AUTOR: NOBOA DELGADO ESTEBAN MARCELO**

**DIRECTOR: MSC. FABIÁN GEOVANNY CUZME RODRÍGUEZ**

**IBARRA – ECUADOR**

**2022**



# 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			
<b>CÉDULA DE IDENTIDAD:</b>	1003869755		
<b>APELLIDOS Y NOMBRES:</b>	Noboa Delgado Esteban Marcelo		
<b>DIRECCIÓN:</b>	Ibarra, Bella vista de Caranqui entre Hualcopo Duchicela y Huiracocha,		
<b>EMAIL:</b>	emnoboat@utn.edu.ec		
<b>TELÉFONO FIJO:</b>		<b>TELÉFONO MÓVIL:</b>	0985245471

DATOS DE LA OBRA	
<b>TÍTULO:</b>	Sistema embebido para reconocimiento y conteo de esporas del hongo de género trichoderma mediante técnicas de visión artificial.
<b>AUTOR (ES):</b>	Esteban Marcelo Noboa Delgado
<b>FECHA: DD/MM/AAAA</b>	05/09/2022
SOLO PARA TRABAJOS DE GRADO	
<b>PROGRAMA:</b>	<input checked="" type="checkbox"/> <b>PREGRADO</b> <input type="checkbox"/> <b>POSGRADO</b>
<b>TÍTULO POR EL QUE OPTA:</b>	Ingeniería en electrónica y redes de comunicación
<b>ASESOR /DIRECTOR:</b>	Ing. Fabián Geovanny Cuzme Rodriguez, MSC.



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**BIBLIOTECA UNIVERSITARIA**

**2. CONSTANCIAS**

El autor manifiesta 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 el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, 05 de septiembre del 2022

**EL AUTOR:**

Esteban Marcelo Noboa Delgado  
CC: 100386975-5



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**BIBLIOTECA UNIVERSITARIA**

**Certificación**

Ing. Fabián Cuzme Rodríguez, MSC., Director del presente trabajo de Titulación

Certifica:

Que el presente trabajo de titulación "Sistema embebido para reconocimiento y conteo de esporas del hongo de género trichoderma mediante técnicas de visión artificial", fue realizado en su totalidad por el Sr. Noboa Delgado Esteban Marcelo, bajo mi supervisión.

Es todo lo que puedo certificar en honor a la verdad.

Ing. Fabián Cuzme Rodríguez, MSC.  
Director



# **UNIVERSIDAD TÉCNICA DEL NORTE**

## **FACULTADA DE INGENIERÍA EN CIENCIAS APLICADAS**

### **DEDICATORIA**

Este trabajo de titulación lo dedico a mi familia, especialmente a mis padres quienes me han brindado su apoyo incondicional e inmutable para lograr esta meta y más por venir, que fueron y son mi sostén moral, emocional y económico en todo esta travesía universitaria y mi vida en general.

Igualmente, a la empresa BiosebOrganics y a sus trabajadores por toda la amabilidad y disposición a ayudar durante la creación del presente proyecto.



# UNIVERSIDAD TÉCNICA DEL NORTE

## FACULTADA DE INGENIERÍA EN CIENCIAS APLICADAS

### AGRADECIMIENTO

Agradezco,

Un especial e infinito agradecimiento a mis padres por toda su confianza, cariño y por estar conmigo tanto en las buenas situaciones como en las malas.

A mi tutor y asesores por su gran trabajo como guías en la realización de esta investigación, haciendo énfasis especial al director del proyecto Ing. Fabián Cuzme por el apoyo científico, moral y legislativo.

A mi familiar y amigo el Ing. Francisco Noboa por su constante apoyo tanto en el ámbito personal como académico.

A la empresa BiosebOrganics por abrirme sus puertas y por su colaboración con asesoría teórica y técnica en la realización del proyecto.

Finalmente, a todos mis amigos cercanos durante este proceso universitario, por sus consejos y los buenos momentos compartidos.

## Tabla de contenido

<b>1. CAPÍTULO I. Antecedentes.....</b>	<b>1</b>
1.1. Tema: .....	1
1.2. Problema:.....	1
1.3. Objetivos .....	3
1.3.1. <i>Objetivo General</i> .....	3
1.3.2. <i>Objetivos Específicos</i> .....	3
1.4. Alcance .....	3
1.5. Justificación .....	4
<b>2. CAPÍTULO II. Revisión bibliográfica .....</b>	<b>6</b>
2.1. Control de plagas.....	6
2.1.1. <i>Plagas en la agricultura</i> .....	7
2.1.1.1. <i>Afectaciones de las plagas en la agricultura</i> .....	7
2.1.2. <i>Control integral de plagas (agentes de bio-control)</i> .....	8
2.2. Hongo de género <i>Trichoderma</i> .....	8
2.2.1. <i>Espora del hongo trichoderma</i> .....	9
2.2.2. <i>Cultivo del hongo de género trichoderma</i> .....	9
2.2.3. <i>Hongo trichoderma en control biológico de cultivos</i> .....	10
2.3. Control de calidad del hongo de género <i>trichoderma</i> .....	11
2.3.1. <i>Determinación del grado de antagonismo</i> .....	11
2.3.2. <i>Determinación de la producción de conidios (esporas)</i> .....	12
2.3.3. <i>Determinación de la viabilidad de conidios (esporas)</i> .....	13
2.3.4. <i>Determinación de contaminantes</i> .....	14
2.3.5. <i>Otras aplicaciones del hongo de género trichoderma</i> .....	15
2.3.5.1. <i>Degradación de contaminantes orgánicos</i> .....	15
2.3.5.2. <i>Degradación de plaguicidas, compuestos organoclorados y compuestos organofosforados</i> .....	16
2.3.5.3. <i>Degradación de Ácido acrílico</i> .....	16
2.3.5.4. <i>Degradación de hidrocarburos del petróleo</i> .....	16
2.4. <i>Visión artificial</i> .....	17
2.4.1. <i>Etapas de la visión artificial</i> .....	17
2.4.2. <i>Niveles de visión artificial</i> .....	18
2.4.3. <i>Aplicaciones de la visión artificial</i> .....	19
2.4.3.1. <i>Medicina</i> .....	19
2.4.3.2. <i>Inspección de calidad</i> .....	20
2.4.3.3. <i>Seguridad basada en video vigilancia</i> .....	20

2.4.3.4.	Visión artificial en biología .....	20
2.4.3.5.	Visión artificial en la industria automotriz .....	21
2.4.3.6.	Visión artificial en rescates .....	21
2.5.	Software para visión artificial .....	21
2.5.1.	Lenguaje de programación Python .....	22
2.5.1.1.	Ventajas de Python .....	22
2.5.2.	Python en aprendizaje de máquinas .....	22
2.6.	Hardware para visión artificial .....	23
2.7.	Aprendizaje de máquina .....	24
2.7.1.	Neurona artificial individual .....	25
2.7.1.1.	Operación para activación de neurona artificial individual .....	25
2.7.1.2.	Error cuadrático medio .....	26
2.7.1.3.	Gradiente descendiente .....	26
2.7.2.	Aprendizaje no supervisado .....	27
2.7.3.	Aprendizaje supervisado .....	27
2.7.4.	Aprendizaje por refuerzo .....	28
2.7.5.	Deep learning .....	28
2.7.5.1.	Red neuronal artificial ANN .....	28
2.7.5.2.	Red neuronal profunda DNN .....	28
2.7.5.3.	Red neuronal recurrente RNN .....	29
2.7.5.4.	Redes neuronales convolucionales CNN .....	29
2.8.	Edge computing .....	30
2.8.1.	Arquitecturas de referencias de Edge Computing .....	30
2.8.2.	Edge Computing 2.0 .....	31
2.9.	Trabajos relacionados .....	32
<b>3.</b>	<b>CAPÍTULO III. Diseño del sistema .....</b>	<b>34</b>
3.1.	Metodología .....	34
3.1.1.	Modelo en cascada .....	34
3.2.	Análisis .....	35
3.2.1.	Técnicas de investigación de recopilación de requisitos .....	35
3.2.1.1.	Cuestionario aplicado al ingeniero encargado .....	36
3.2.1.2.	Revisión de trabajos relacionados .....	37
3.2.1.3.	Mesa de trabajo .....	38
3.3.	Análisis de las técnicas de recolección de información .....	39
3.3.1.	Análisis del cuestionario .....	39
3.3.2.	Análisis de la revisión de trabajos relacionados .....	41

3.3.3.	<i>Análisis de la mesa de trabajo</i> .....	41
3.4.	Requerimientos del sistema .....	42
3.4.1.	<i>Beneficiarios</i> .....	42
3.4.1.1.	Requerimientos de stakeholders .....	44
3.4.1.2.	Requerimientos de hardware y software del sistema .....	46
3.5.	Recursos .....	48
3.5.1.	<i>Selección de hardware</i> .....	49
3.5.1.1.	Placa embebida .....	49
3.5.1.2.	Cámara con acople a microscopio .....	50
3.5.1.3.	Protección contra manipulación y clima artificial.....	50
3.5.2.	<i>Selección de software</i> .....	53
3.5.2.1.	Lenguaje de programación de alto nivel Python .....	53
3.5.2.2.	Tipo de aprendizaje profundo (Deep Learning).....	54
3.5.2.3.	Plataforma en nube compatible con Edge Computing.....	54
3.6.	Diseño del sistema.....	55
3.6.1.	<i>Diagrama y flujograma del sistema</i> .....	55
3.6.1.1.	Sistema de entrenamiento .....	57
3.6.1.2.	Sistema de conteo de esporas trichoderma .....	62
3.7.	Costos asociados al desarrollo del prototipo.....	69
4.	<b>CAPÍTULO IV. Implementación y pruebas de funcionamiento</b> .....	72
4.1.	Código del sistema .....	72
4.1.1.	<i>Código del sistema de entrenamiento</i> .....	72
4.1.1.1.	Sincronizar base de datos.....	73
4.1.1.2.	Proceso de entrenamiento .....	75
4.1.2.	<i>Código del sistema de conteo</i> .....	77
4.1.2.1.	Adquisición de imagen .....	78
4.1.2.2.	Especificaciones de imagen y aprendizaje profundo .....	79
4.1.2.3.	Edición imagen y matriz de detección de esporas .....	80
4.1.2.4.	Función para resaltar objetos encontrados.....	81
4.1.2.5.	Conexión hacia nube IoT.....	83
4.2.	Validación de requerimientos de Stakeholder .....	84
4.3.	Pruebas funcionales.....	85
4.3.1.	<i>Pruebas individuales interfaz gráfica</i> .....	85
4.3.1.1.	Pruebas funcionales de la interfaz gráfica .....	85
4.3.1.2.	Pruebas usabilidad de interfaz gráfica.....	87
4.3.1.3.	Pruebas validación de interfaz gráfica .....	88

4.2.1.	<i>Pruebas de aplicativo IoT</i> .....	88
4.2.1.1.	<i>Pruebas de envío de datos</i> .....	89
4.2.1.2.	<i>Pruebas de almacenamiento de datos</i> .....	90
4.2.1.3.	<i>Pruebas de visualización</i> .....	91
4.2.2.	<i>Case protector</i> .....	92
4.3.	<i>Pruebas funcionales locales</i> .....	93
4.4.	<i>Pruebas de producción</i> .....	94
4.4.1.	<i>Primera semana de pruebas</i> .....	96
4.4.2.	<i>Segunda semana de pruebas</i> .....	97
4.4.3.	<i>Tercera semana de pruebas</i> .....	98
4.5.	<i>Validación y métrica de eficiencia</i> .....	99
4.5.1.	<i>Métrica de eficiencia</i> .....	99
4.5.1.2.	<i>Matriz de confusión</i> .....	99
4.5.1.3.	<i>Métricas de eficiencia pruebas de funcionamiento</i> .....	100
4.5.1.4.	<i>Métricas de eficiencia pruebas de producción</i> .....	100
4.5.1.5.	<i>Promedio de la métrica precisión del sistema</i> .....	104
4.6.	<i>Costo/Beneficio</i> .....	105
	<i>Conclusiones</i> .....	110
	<i>Recomendaciones</i> .....	112
	<b>ANEXOS: .....</b>	<b>113</b>
	<b>ANEXO A: Carta de colaboración de empresa BiosebOrganics</b> .....	<b>113</b>
	<b>ANEXO B: Formato de cuestionario con resultados</b> .....	<b>114</b>
	<b>ANEXO C: Revisión de trabajos relacionados</b> .....	<b>118</b>
	<b>ANEXO D: Mesa de trabajo</b> .....	<b>121</b>
	<i>i. Análisis del aumento y recolección de requisitos</i> .....	128
	<i>ii. Definición de la importancia de cada requisito</i> .....	129
	<b>ANEXO E: Código del sistema</b> .....	<b>134</b>
	<i>iii. Código de entrenamiento</i> .....	134
	<i>iv. Código del sistema de detección y conteo</i> .....	134
	<b>ANEXO F: Lista de asistencia a capacitación</b> .....	<b>139</b>
	<b>REFERENCIAS</b> .....	<b>140</b>

## Índice de figuras

Figura 1 Esporas del hongo trichoderma bajo microscopio	9
Figura 2 Técnica del cultivo dual con Trichoderma (Tr) y patógeno Colletotrichum (Co), siendo un grado 3 en la escala de BELL	11
Figura 3 Conteo microscópico de esporas del hongo de género trichoderma	13
Figura 4 Esquema neurona artificial individual	25
Figura 5 Gradiente en gráfico	26
Figura 6 Compuesto de esporas trichoderma sin disolución	40
Figura 7 Parte superior del case de protección	51
Figura 8 Elemento de case inferior	52
Figura 9 Proceso de impresión en 3D	52
Figura 10 Diagrama de bloques general del sistema	55
Figura 11 Aplicación LabelImg	58
Figura 12 Archivo de etiquetado classes.txt	58
Figura 13 Contenido de archivos etiquetación	58
Figura 14 Diagrama bloque entrenamiento	59
Figura 15 Diagrama flujo para el entrenamiento del sistema	60
Figura 16 Página de inicio de plataforma IoT	64
Figura 17 Diagrama de bloque de funcionamiento	65
Figura 18 Flujograma sistema de conteo y detección	67
Figura 19 Tarjeta gráfica de Google Colab	73
Figura 20 Montaje directorio de Google Drive	73
Figura 21 Permisos para uso de Google Drive	73
Figura 22 Cambio de directorio	74
Figura 23 Guardar ruta de base de datos de entrenamiento en variables	74
Figura 24 Modificación de archivo de configuración	75
Figura 25 Archivo comprimido de Darknet	75
Figura 26 Exportar directorio que contiene el código Darknet	75
Figura 27 Descomprimir archivo descargado	76
Figura 28 Comando para entrenamiento	76
Figura 29 Archivos espora.data y espora.cfg	77
Figura 30 Selector de imagen	77
Figura 31 Código función de interfaz gráfica	78
Figura 32 Código de especificación de imagen y aprendizaje profundo	80
Figura 33 Código para función de edición de imagen y matriz de detección	81
Figura 34 Código para la función resaltar objetos	82
Figura 35 Código de función para subir información a nube	83
Figura 36 Primera vista aplicación gráfica del sistema de detección y conteo de esporas trichoderma	85
Figura 37	85
Figura 38 Imagen seleccionada de entrada	86
Figura 39 Detección de esporas en interfaz gráfica	86
Figura 40	88
Figura 41 Trama respuesta DNS	89
Figura 42 Establecimiento de conexión entre el sistema y la plataforma IoT	89
Figura 43 Respuesta de protocolo TSL del servidor	90
Figura 44 Guardado de datos	90
Figura 45 Representación con tecnología Edge Computing	92

Figura 46 Resultado final del almacón	93
Figura 47 Fotografía de la inducción al personal	94
Figura 48 Fotografía del sistema en su ambiente de producción	95
Figura 49 Mapa resultante de lluvia de ideas, ideas principales	124
Figura 50 Hardware para adquisición de imágenes y placas embebidas	125
Figura 51 Software para tratamiento de imágenes y lenguaje de programación	126
Figura 52 Técnicas de visión artificial	127

## Índice de tablas

Tabla 1 Escala de Bell	11
Tabla 2 Preguntas del cuestionario	36
Tabla 3 Ajuste ficha bibliográfica 1	37
Tabla 4 Descripción de acrónimos	42
Tabla 5 Tabla de beneficiarios	42
Tabla 6 Especificaciones para tabla de requerimientos	43
Tabla 7 Tabla de requisitos de usuario	44
Tabla 8 Tabla de requisitos iniciales del sistema	44
Tabla 9 Tabla de requisitos funcionales del sistema	46
Tabla 10 Recursos humanos	48
Tabla 11 Cuadro comparativo placas embebidas	49
Tabla 12 Cuadro comparativo cámaras digitales	49
Tabla 13 Cuadro comparativo protección externa	50
Tabla 14 Cuadro comparativo lenguajes de programación	53
Tabla 15 Cuadro comparativo tipos de algoritmo de aprendizaje profundo	54
Tabla 16 Listado recursos de plataformas Edge Computing	54
Tabla 17 Componentes de software	69
Tabla 18 Componentes físicos	70
Tabla 19 Validación de requerimientos de Stakeholders	84
Tabla 20 Tareas para usabilidad del sistema	87
Tabla 21 Pruebas de validación	88
Tabla 22 Propiedades mecánicas generales PLA	92
Tabla 23 Comparación de conteo automático del sistema y conteo manual a concentración (1/10) ml	93
Tabla 24 Comparación de conteo automático del sistema y conteo manual a concentración (1/10) ml en pruebas de producción, primera semana	96
Tabla 25 Comparación de conteo automático del sistema y conteo manual a concentración (1/100) ml en pruebas de producción, primera semana	96
Tabla 26 Comparación de conteo automático del sistema y conteo manual a concentración (1/10) ml en pruebas de producción, segunda semana	97
Tabla 27 Comparación de conteo automático del sistema y conteo manual a concentración (1/100) ml en pruebas de producción, segunda semana	97
Tabla 28 Comparación de conteo automático del sistema y conteo manual a concentración (1/10) ml en pruebas de producción, tercera semana	98
Tabla 29 Comparación de conteo automático del sistema y conteo manual a concentración (1/100) ml en pruebas de producción, tercera semana	98
Tabla 30 Matriz de confusión de las pruebas de funcionamiento	99
Tabla 31 Matriz de confusión de las pruebas de producción en disolución de 1ml a 10ml de la primera semana de muestreo	100
Tabla 32 Matriz de confusión de las pruebas de producción en disolución de 1ml a 100ml de la primera semana de muestreo	101
Tabla 33 Matriz de confusión de las pruebas de producción en disolución de 1ml a 10ml de la segunda semana de muestreo	101
Tabla 34 Matriz de confusión de las pruebas de producción en disolución de 1ml a 100ml de la segunda semana de muestreo	102
Tabla 35 Matriz de confusión de las pruebas de producción en disolución de 1ml a 10ml de la tercera semana de muestreo	102

Tabla 36 Matriz de confusión de las pruebas de producción en disolución de 1ml a 100ml de la tercera semana de muestreo	103
Tabla 37 Promedio de métricas del sistema	104
Tabla 38 Costos de hardware asociados al desarrollo del prototipo	105
Tabla 39 Costos de software asociados al desarrollo del prototipo	106
Tabla 40 Costos de ingeniería	107
Tabla 41 Costos totales	108
Tabla 42 Tabla Costo/Beneficio	109
Tabla 43 Cronograma de la actividad mesa de trabajo	122
Tabla 44 Especificaciones para tabla de requerimientos	129
Tabla 45 Análisis requisitos para placa embebida	129
Tabla 46 Análisis requisitos para cámara con acople a microscopio	130
Tabla 47 Análisis de requisitos de protección externa del sistema	131
Tabla 48 Análisis requisitos para lenguaje de programación	131
Tabla 49 Análisis requisitos para lenguaje de programación	132
Tabla 50 Análisis requisitos para plataforma Edge Computing	133

## **1. CAPÍTULO I. Antecedentes**

El capítulo de antecedentes tiene como objetivo establecer las razones que impulsaron el desarrollo del proyecto de investigación presentando la problemática, los objetivos generales y específicos, el alcance y la justificación de los fundamentos del trabajo de grado. Para lograr este fin, se utiliza la metodología PMBOK que tiene entre sus objetivos el obtener estos conocimientos, aquí se hace uso específicamente del proceso de gestión integral de proyectos en lo concerniente a los subprocesos de acta de constitución y desarrollo del plan de proyecto. Además, se empleó el proceso de gestión de alcance, su definición y la planificación del cronograma incluyendo los subprocesos de creación de la EDT/WBS.

### **1.1. Tema:**

Sistema embebido para el reconocimiento y conteo de esporas de hongo trichoderma basado en técnicas de visión artificial.

### **1.2. Problema:**

El género *Trichoderma* es un hongo cosmopolita, esta denominación se aplica a los organismos que pueden encontrarse en cualquier territorio del planeta, pero solo en condiciones propicias para su subsistencia, esto no implica que ocupe todos los rincones de estos sectores (Fumanal, 2013). Según Bustillo en su publicación científica “Prevención de los riesgos para la salud derivados del uso de plaguicidas en la agricultura” (2004): los órganos rectores en varias partes del mundo implementan mandatos legislativos con el objetivo de disminuir la dependencia de los plaguicidas en la agricultura para aumentar la seguridad ambiental y de los consumidores (Bustillo, 2004). Con el fin de reducir los riesgos asociados con las aplicaciones de plaguicidas y reducir la dependencia de su uso, un enfoque incluye el uso de agentes de control biológico y sus productos como alternativas a los agroquímicos sintéticos, específicamente los hongos de género *Trichoderma* que se encuentran entre los agentes de control biológico microbiano (MBCA) más utilizados en la agricultura (Woo et al., 2014).

La empresa BiosebOrganics dedicada a la elaboración de bioproductos basados en microorganismos orientados a la agricultura, posee una propiedad inmueble que trabaja como sucursal situada en la ciudad de Ibarra en las calles José Mejía Lequerica

8-33 y Manuel de la Chica Narváez. En esta sucursal la empresa tiene tres áreas que son bodega, envasadora y la zona de cultivo del microorganismos (Rodríguez, 2011).

La línea de producción en el cultivo de microorganismos tiene un cuello de botella en la etapa de verificación de calidad, debido a que se realiza el conteo usando un microscopio adecuado (cabe recalcar que una muestra puede contener cientos de esporas o cualquier otro microorganismo), en este caso esporas del hongo trichoderma y si existieran elementos intrusos en la mezcla (bacterias como cocos, bacillus, entre otros), este proceso se repite con varias muestras de un mismo cultivo con el fin de constatar la cantidad de hongo trichoderma criado (calidad del cultivo). Este proceso consume mucho tiempo en el ciclo de producción, ya que para realizar una observación de esporas de este hongo se requiere tratar la muestra con el método denominado: “azul de lactofenol” que es un preparado con propiedades de colorante simple utilizado en laboratorios clínicos para colorear principalmente estructuras fúngicas como esporas (Gil, 2019). Posteriormente para un conteo más preciso de esporas de trichoderma se utiliza la técnica de hemocitómetro (Estrada et al., 2002) o cámara de Neubauer que se basa en un artefacto de vidrio que tiene dos superficies reticuladas y dos columnas laterales con una altura de 0,1mm arriba del retículo; el retículo de Neubauer tiene una subdivisión en 9 cuadrantes, de estos resultan útiles los 4 cuadrantes laterales de L1, L2, L3 y L4 estos junto con el cuadrante central, que a su vez se subdivide en varios pequeños sub-cuadrantes, con un total de 25 H (Oliveira, 2019). Este artefacto ayuda a la persona que cuenta microorganismos a no equivocarse ya que divide a un portaobjetos en cuadrículas microscópicas como se especificó.

Para brindar una solución a lo planteado se propone la implementación de un sistema embebido que albergue un algoritmo de Deep Learning aplicado a visión artificial, permitiendo identificar y contar esporas del hongo de género Trichoderma existentes en una imagen tomada mediante una cámara digital acoplada a un microscopio con el fin de brindar una cifra exacta de la cantidad de esporas y figuras que no concuerden (pueden ser bacterias bacillus, cocos entre otros).

### **1.3. Objetivos**

#### **1.3.1. Objetivo General**

Desarrollar un sistema de identificación y conteo de esporas del hongo de género trichoderma basado en técnicas de visión artificial con el fin de contribuir al proceso de control de calidad en la crianza de este microorganismo.

#### **1.3.2. Objetivos Específicos**

- Revisar el estado del arte en fuentes fiables y actuales que permitan abordar el tema de investigación fuera de la ingeniería y fomentar las bases del proyecto.
- Diseñar el sistema de identificación y conteo de esporas por métodos de visión artificial tomando en cuenta tanto el hardware como el software del sistema para un correcto funcionamiento y una forma amigable de observar los resultados.
- Realizar pruebas de funcionamiento, donde se enfocará en la calidad de imágenes obtenidas y si este algoritmo de Deep Learning brinda resultados con un margen de error aceptable y acorde a los obtenidos en el conteo en placa Petri.
- Realizar pruebas de producción con enfoque en el registro de los resultados brindados por el sistema de identificación y conteo de esporas del hongo trichoderma, en un periodo de prueba en los laboratorios de la sede Ibarra de la empresa BiosebOrganics.

### **1.4. Alcance**

El presente proyecto aborda conceptos de crianza de hongos del género trichoderma en un ambiente controlado. Las etapas de este proceso incluyen el control de calidad dónde se empleará el sistema de identificación y conteo de esporas del hongo Trichoderma, así se obtendrá la información referente a la cantidad de esporas en una muestra observada a través de un microscopio y capturada con una cámara digital mediante técnicas de visión artificial. El algoritmo de Deep Learning se estructurará con una base de datos de imágenes previas para hacer una comparación con la nueva muestra suministrada, cabe recalcar que al usar una cámara digital de mayor calidad en cuanto a píxeles se obtendrá mayor precisión.

El diseño del presente proyecto se realiza en base al modelo para dirección de proyectos PMBOK, principalmente el modelo de tiempo de vida en dónde se observan cinco procesos: inicio, planificación, control y monitorización, ejecución y cierre (Project Management Institute, 2017). Para el diseño del sistema y sus algoritmos

enfocados en visión artificial se utiliza el modelo en espiral para desarrollo de software, debido a que cuenta con distintas etapas que van desde la conceptualización, el desarrollo, una fase de mejoras y finalmente el mantenimiento (la cual no se toma en cuenta para el proyecto), etapas muy útiles para el desarrollo del algoritmo de deep learning y el armado del hardware del sistema (Montero et al., 2018).

El algoritmo de Deep Learning se desarrollará en el lenguaje de alto nivel python por su amplia selección de librerías y frameworks referentes a visión artificial y Deep Learning (Buleje et al., 2021). En la ejecución de las pruebas de funcionamiento de este algoritmo se comparará la matriz de resultados de la imagen capturada con los diferentes matices de aprendizaje ubicadas en una tarjeta Micro SD de 128GB en la placa embebida Jetson Nano, con el fin de obtener la identificación de la forma de las esporas del hongo Trichoderma y para su posterior conteo, separando figuras que no coinciden y contabilizándolas. Posteriormente se publican los resultados en una plataforma en la nube utilizando la tecnología IoT Edge Computing.

Para finalizar, se adjunta fotografías del sistema embebido ya en ejecución, brindando tablas de resultados de pruebas de funcionamiento individual del hardware del sistema embebido, del algoritmo de visión artificial y del sistema en su totalidad. Además, se añade la documentación de las pruebas del proceso de validación del dispositivo en producción en un periodo de tiempo, con el fin de dar posibilidad a reajustes.

## **1.5. Justificación**

Los microorganismos de ciertos tipos, en agricultura son de suma importancia para conservar la fertilidad del suelo y control de plagas para obtener cultivos sanos y de buena calidad. Los hongos de género trichoderma están ganando popularidad gracias a resultados positivos en su aplicación como alternativa al uso de otro tipo de fertilizantes. (Terou & Parr, 1967).

Según el Plan del Buen Vivir creado por el Consejo Nacional de Planificación correspondiente al período 2017 - 2021; el sector agrícola en el cual se pretende aplicar el presente proyecto pertenece a las áreas denominadas de investigación prioritaria y tomando como referencia el objetivo número 6 el cual especifica que: "Desarrollar las Capacidades Productivas y del Entorno para lograr la Soberanía Alimentaria y el

Desarrollo Rural Integral". Es así como, de acuerdo con lo expuesto, el presente proyecto pretende apoyar la producción agroecológica. Los productos basados en el hongo *Trichoderma* cumplen una importante función en la agricultura ecológica puesto que detienen el crecimiento y propagación de ciertos hongos dañinos para los cultivos evitando el uso de fungicidas y otros químicos perjudiciales para la salud humana y medio ambiente (Consejo Nacional de Planificación, 2017).

Adicionalmente, se cuenta con el apoyo de la empresa BiosebOrganics que trabaja produciendo microorganismos con enfoque en la agricultura, aquí se dispone de la información requerida, apoyo logístico y monetario para el desarrollo del proyecto. Donde el trabajo resultante brindará una forma más rápida y eficaz de conteo de esporas del hongo *trichoderma* y demás elementos intrusos en una muestra observada a través de un microscopio, mediante un sistema embebido que consta de una cámara digital con acople hacia microscopio y un ordenador de placa reducida "Jetson Nano" y un algoritmo de visión artificial basado en Deep Learning. Esto con el fin de obtener mayor calidad de la muestra basado en la cantidad de esporas reconocidas y agentes intrusos encontrados. La información necesaria para el desarrollo del proyecto, específicamente en el tema acerca del hongo de género *trichoderma*, es brindado por la empresa BiosebOrganics. En cuanto al conocimiento de ingeniería en la rama de Electrónica y Redes de Comunicación hace referencia al funcionamiento del sistema embebido tanto la cámara digital como de la placa (ordenador reducido "Nvidia Jetson Nano") y al algoritmo de visión artificial, en sus pruebas tanto individuales como colectivas.

Se crea un impacto económico ya que, ahorra tiempo en el proceso de elaboración de bioproductos basados en el hongo *trichoderma* de la empresa BiosebOrganics, específicamente en el control de calidad. Debido a que, en el conteo de esporas del hongo se crea un cuello de botella al ser realizado de manera manual, lo cual representa alto consumo de tiempo y recursos. En cuanto al impacto social, los productos elaborados por la empresa tienen como objetivo el control de plagas con productos alternativos a los fungicidas altamente perjudiciales para el ambiente, trabajadores e incluso comunidades cercanas (Bustillo, 2004). Adicionalmente se colocó un impacto científico académico ya que se propone una aplicación innovadora de algoritmos de aprendizaje profundo con un enfoque en visión artificial, como es el reconocimiento y conteo de microorganismos microscópicos como son las esporas del hongo *trichoderma*.

## 2. CAPÍTULO II. Revisión bibliográfica

En el capítulo dos se presenta el marco teórico el cual contiene la temática necesaria para la comprensión y desarrollo del proyecto. Para su estructura se hizo uso de la metodología PMBOK específicamente en el proceso de gestión de integración y los subprocesos de gestión de conocimiento. Se da inicio con la introducción referente al control de plagas para dar paso al enfoque de control integral de plagas dentro de la agricultura. Posteriormente se señala conceptos referentes al hongo trichoderma, su aplicación en el control integral de plagas junto con otras aplicaciones y las técnicas de control de calidad en el proceso de crianza de este hongo. La segunda parte consta de los temas de visión artificial y su proceso de ejecución, software y hardware para el armado del sistema, aquí se especifican conceptos de hardware y del lenguaje de programación (Python) a utilizar para el desarrollo del programa. Adicional a esto, se aborda información de técnicas de aprendizaje profundo, neurona artificial individual y conceptos de Edge computing. Finalmente, se cita trabajos relacionados o similares al tema de investigación con enfoque en visión artificial dentro de la biología.

### 2.1. Control de plagas

El control de plagas es una actividad de protección de la salud humana y animal que trata de impedir o disminuir la probabilidad de padecer un daño determinado por la acción de cualquier organismo nocivo (Pinto Rodríguez, 2018).

Las plagas pueden causar infecciones, transmitir enfermedades graves, alterar el medioambiente, provocar pérdidas económicas, causar problemas psicológicos, proliferándose más fácilmente cuando consiguen el alimento, el agua o el refugio imprescindibles para la supervivencia y desarrollo biológico de la plaga.

El control de plagas hace uso de productos denominados plaguicidas, palabra que se deriva del latín “cida” que significa exterminador, así tenemos diferentes tipos de plaguicidas que se emplean contra diferentes plagas en la agricultura o el hogar (Alfáu Ascuasiati, 2012). A continuación, se observan diferentes ejemplos:

- Biocidas: sustancia empleada para eliminar organismos vivos en el general.
- Insecticidas: para plagas de insectos sea en agricultura o en el hogar.
- Fungicidas: para plagas de hongos sea en agricultura o en el hogar.

- Herbicidas: para plagas de hierbas.
- Raticidas: para plagas de ratas sea en agricultura o en el hogar.
- Bactericidas: para plagas de bacterias sea en agricultura o en el hogar.

En el hogar también es necesario el control de plagas debido a que existen organismos dañinos para sus habitantes podemos citar los más frecuentes: ácaros, gorgojo, polillas, cucarachas, garrapatas, pulgas, ratas entre otros (Elorza, 2016).

### ***2.1.1. Plagas en la agricultura***

Según la OMS (Organización Mundial de la Salud) “las plagas son organismos que causan un daño inaceptable a cultivos o productos almacenados, que amenazan la salud humana o animal, siendo objeto de tratamientos de protección vegetal, de salud pública o de productos domésticos. Para la FAO (Organización de Naciones Unidas para la Agricultura y Alimentación) las plagas son cualquier especie, raza, biotipo vegetal o animal o agente patógeno, dañina para las plantas o productos vegetales (OMS & FAO, 2010). Una plaga no es el virus o bacteria en sí misma, sino la situación en la que un organismo vivo (patógeno) ocasiona alteraciones fisiológicas en otro, normalmente con síntomas visibles o daños económicos.

Posteriormente se revisa información acerca del umbral de tolerancia para plagas, este se considera como la densidad de población de una especie en donde su presencia ya se considera un riesgo para la salud, confort, higiene o la seguridad de las personas y animales; se podría decir que esto provoca implicaciones económicas, sanitarias o ambientales inaceptables (García Laureano, 2020).

#### **2.1.1.1. Afectaciones de las plagas en la agricultura.**

Los daños producidos por las plagas pueden clasificarse en económicos, relacionados con la salud y relacionados con el bienestar (García Laureano, 2020). Principalmente hay dos tipos de plaga en relación con el daño ocasionado y la parte vegetal afectada:

- **Plagas directas:** dañan los órganos de la planta que van a ser cosechadas, por ejemplo, el pulgón.
- **Plagas indirectas:** si se daña otras partes de la planta que no son cosechables, como los mosquitos minadores que dañan las hojas del tomate cuando lo que se recolecta es el fruto (Maya Álvarez, 2017).

### **2.1.2. Control integral de plagas (agentes de bio-control)**

El control de plagas convencional emplea métodos químicos para luchar contra diferentes tipos de plagas, haciendo uso indiscriminado de biocidas y productos fitosanitarios inclusive de forma preventiva, sin tener en cuenta la huella medioambiental o la toxicidad que pueda alcanzar sobre la salud de la población tanto humana como animal.

El control integral de plagas es un sistema de gestión, que teniendo en cuenta el medioambiente, las especies dañinas y las útiles se usa todas las técnicas disponibles de la forma más compatible posible, manteniendo a las especies nocivas en niveles inferiores a los que causan daños económicos o pérdidas inaceptables (Pinto Rodríguez, 2018). Una de las técnicas amigables con el medio ambiente empleada es la utilización del hongo de género trichoderma para el control de ciertas plagas.

Los agentes de biocontrol para control integral de plagas tienen sus bases en el desarrollo de biofungicidas o productos biológicos que actúan como fungicidas. La materia activa de un gran parte de estos productos es un hongo o una bacteria. Se pueden citar varias materias activas como agentes de biocontrol: bacterias *Agrobacterium radiobacter* (Beijerinck y van Delden), hongos filamentosos donde pertenecen diferentes especies de hongos de género trichoderma (Jacas et al., 2014).

## **2.2. Hongo de género Trichoderma**

Según Alexopoulos C. Mims C. y Blackwell M. (2018) Las especies del género *Trichoderma* representan un grupo de hongos filamentosos de pertenencia al Reino Mycetae (fungi), división Eumycota, subdivisión Deuteromycotina, clase Hyphomycetes, orden Hyphales (Moniliales) y familia Moniliaceae (Alexopoulos et al., 1996). Aunque de acuerdo con Sánchez Hernández, Rosa, Rosique Gil, & Pachero Figueroa (2018) la clasificación taxonómica del género *Trichoderma* sería Reino Mycetae (Fungi), División Eumycota, subdivisión Ascomycotina, clase Eucomycetes, orde Hypocreales, Familia Hypocraceae y género *Trichoderma* y *Hypocrea* (Sánchez Hernández et al., 2018).

Estos hongos se identifican por predominar en ecosistemas terrestres (suelos agrícolas, pastizales, bosques y desiertos) y acuáticos (Argumedo Delira et al., 2009). Algunas especies son de vida libre en el suelo, oportunistas, simbioses de plantas, y otras

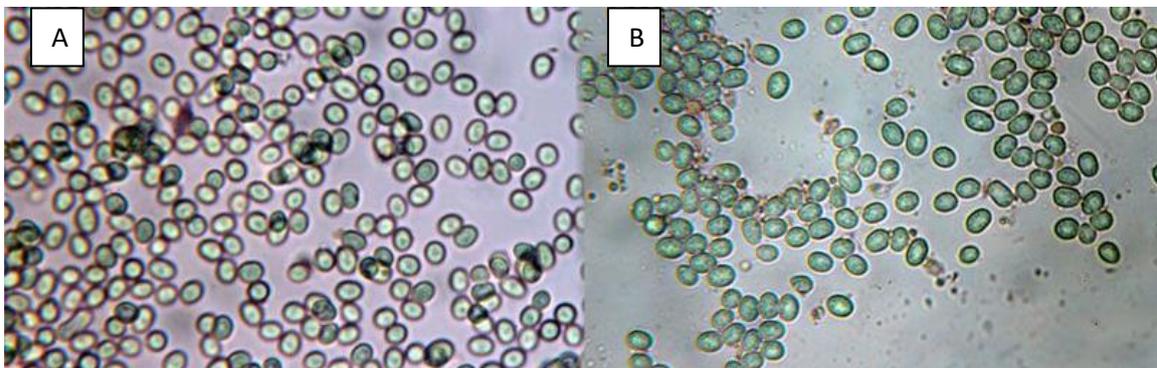
son mico parásitas. También son capaces de colonizar distintos tipos de ambientes, gracias a su alta capacidad reproductiva (Bissett, 1991).

### 2.2.1. Espora del hongo trichoderma

**La espora es una célula reproductiva** propia de algunas plantas y hongos con la capacidad de dividirse sucesivamente para acabar formando un nuevo ser, descendiente de su progenitor. Son consideradas como un tipo de **reproducción asexual debido a que** no necesitan ningún otro ente para dividirse y formar una nueva planta u hongo. A continuación, observamos en la Figura 1 diferentes tomas de esporas del hongo de género trichoderma, con un escrito de su tamaño aproximado dando como un promedio las medidas de: 3,3x2,3 micro metros (también se muestra un rango de 3,1 a 3,6  $\mu\text{m}$  de largo y 2,2 a 2,4  $\mu\text{m}$  de ancho).

Figura 1

*Esporas del hongo trichoderma bajo microscopio*



*Nota: Tanto el cuadro A como el cuadro B contienen esporas de hongo trichoderma con una medida que va entre 3,1 a 3,6  $\mu\text{m}$  de largo y 2,2 a 2,4  $\mu\text{m}$  de ancho, tomando en cuenta el promedio de medida 3,3  $\mu\text{m}$  de largo x 2,3  $\mu\text{m}$  de ancho.*

*Fuente: (Castillo, 2015)*

### 2.2.2. Cultivo del hongo de género trichoderma

El cultivo se realiza mediante el proceso de fermentación de un medio líquido o sólido ubicado en biorreactores o matrices dentro de las instalaciones o laboratorios especializados. La técnica de fermentación es el proceso por el cual los microorganismos catalizan nutrientes, sintetizan metabolitos secundarios y completan otras actividades fisiológicas en condiciones anaeróbicas o aeróbicas. En esta etapa, se acumulan los microorganismos o metabolitos microbianos deseados. Los microorganismos

comprometidos en este proceso de fermentación, obtienen su energía y fuente de carbono mediante la degradación de compuestos orgánicos (Gómez, 2017). A continuación, se observa diferentes tipos de procesos de fermentación:

- **Fermentación en estado sólido:** La fermentación sólida es un proceso que se identifica por el crecimiento microbiano en un sustrato sólido (maíz, arroz, bagazo de caña de azúcar, residuos de trigo, plátano, etc.) con el objetivo de optimizar la cantidad de proteínas y otros nutrientes (minerales, grasas, carbohidratos y vitaminas) presentes en un alimento (Herrera Beltran, 2021).
- **Fermentación en estado líquido:** Normalmente, el proceso de fermentación inicia con un cultivo de arranque puro que se cultiva en placas de Petri o en medio líquido en un matraz de agitación. Para el proceso de fermentación discontinua, se inocular asépticamente un lote de medio de cultivo líquido (como melazas) con un microorganismo que se desea producir, esto se denomina cultivo de semilla o de arranque. Dónde con las mejores condiciones de fermentación, se conservan durante una duración definida para el cultivo máximo de microorganismos, en un recipiente cerrado denominado fermentador o biorreactor (Gómez, 2017). Una fermentación discontinua se define como un "sistema cerrado". Al comienzo de los procesos se coloca una solución esterilizada de nutrientes y se inocular con el microorganismo, permitiendo que se lleve a cabo la incubación en condiciones óptimas de fermentación. La solución de nutrientes, en la fermentación en estado líquido, debe ser líquido pudiendo ser cercana a ser viscosa (Reyes, 2017).

### ***2.2.3. Hongo trichoderma en control biológico de cultivos***

El control biológico es la utilización de organismos enemigos naturales de una plaga o patógeno, con el objetivo de reducir o eliminar sus efectos dañinos en las plantas o sus productos (Muñoz Cuaical, 2016). Los productos fungicidas biológicos en base de microorganismos benéficos del género *Trichoderma* ayudan a resguardar a las plantas de diversas enfermedades del suelo y también logran nutrir las raíces para un óptimo desarrollo de los cultivos, estos productos biológicos hacen que se reduzca el uso de pesticidas químicos, perseverando la salud de seres humanos y evitan la contaminación del ambiente y suelo (Infante et al., 2019).

### 2.3. Control de calidad del hongo de género trichoderma

El control de calidad del producto final es uno de los elementos del proceso más importante debido a que certifica el funcionamiento y su tolerancia a largo plazo. Al tratarse de productos con base en microorganismos se requiere instrumentos de laboratorio calificado para este tipo de procesos. Se prosigue a describir los tipos de pruebas de calidad:

- Determinación del grado de antagonismo.
- Determinación de la producción de conidios.
- Determinación de la viabilidad de los conidios.
- Determinación de contaminantes (Pureza).

#### 2.3.1. Determinación del grado de antagonismo

Se inicia con la técnica de cultivo dual, usualmente en placas de Petri con medio APG (Agar Papa Glucosado al 2%) aquí se depositan discos de micelio (5 mm) de cada aislamiento de Trichoderma y del fitopatógeno (microorganismos plaga que es atenuado por el hongo trichoderma) en posiciones opuestas y equidistantes como se muestra en Figura 2. Las pruebas se realizan por triplicado. Luego de siete días de incubación a 25° C se evalúa el grado de antagonismo utilizando la escala de Bell mostrada en la Tabla 1.

Tabla 1

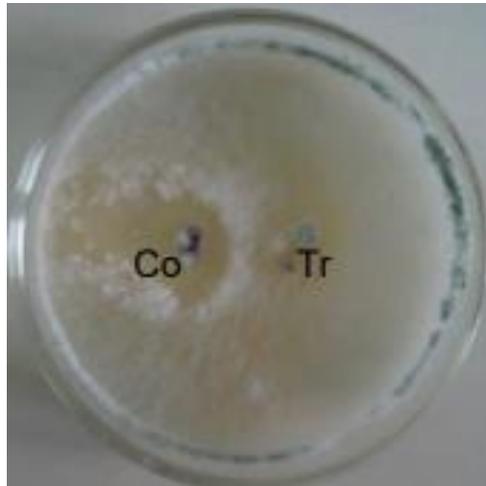
Escala de Bell

Núm.	Capacidad antagónica
1	El antagonista crece completamente sobre el patógeno y cubre totalmente la superficie del medio de cultivo
2	El antagonista crece las dos terceras partes de la superficie del medio de cultivo
3	Antagonista y patógeno colonizan la mitad de la superficie del medio de cultivo y ninguno de los dos domina sobre el otro
4	El patógeno coloniza al menos las dos terceras partes de la superficie del cultivo
5	El patógeno crece por encima del antagonista y ocupa casi toda la superficie del medio

Fuente: (Sandoval & Belesansky, 2020)

Figura 2

Técnica del cultivo dual con *Trichoderma* (Tr) y patógeno *Colletotrichum* (Co), siendo un grado 3 en la escala de BELL



Fuente: (Sandoval & Belesansky, 2020)

### 2.3.2. Determinación de la producción de conidios (esporas)

Este proceso implica tomar diferentes muestras de la mezcla resultante en la crianza (en cajas Petri o en biorreactores), se coloca sobre el portaobjetos y se cubre con la pieza de vidrio denominada cubreobjetos, para el examen minucioso en un microscopio donde se cuenta las esporas del hongo perteneciente a trichoderma. Finalmente se realiza una ecuación simple para observar el porcentaje total de esporas de hongos de género trichoderma en relación con impurezas, como pueden ser bacterias cocos o bacillus muy comunes en el medio ambiente (Silva & Alvarez, 2013). Este proceso se repite con varias muestras de un mismo cultivo con el fin de constatar la cantidad de hongo trichoderma criado mediante la cantidad de conidios que este produce. (Silva & Alvarez, 2013). Las técnicas de conteo de esporas que se pueden considerar son:

- **Por coloración**

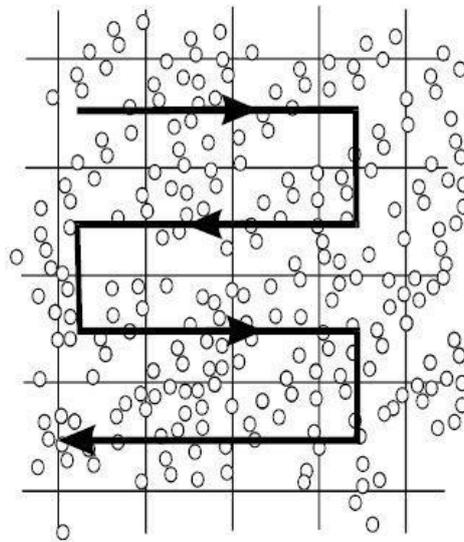
Para realizar una observación de las esporas de este tipo de hongo se requiere tratar la muestra con el método denominado “azul de lactofenol”, que es un preparado con propiedades de colorante simple utilizado en laboratorios clínicos para colorear principalmente estructuras fúngicas como esporas (Gil, 2019).

- **Técnica de hemocitómetro o cámara de Neubauer**

Esta técnica se basa en un artefacto de vidrio que tiene dos superficies reticuladas y dos columnas laterales con una altura de 0,1mm arriba del retículo; el retículo de Neubauer tiene una subdivisión en 9 cuadrantes, de estos resultan útiles los 4 cuadrantes laterales de L1, L2, L3 y L4 estos junto con el cuadrante central, que a su vez se subdivide en varios pequeños sub cuadrantes, con un total de 25 H. Este artefacto ayuda a la persona que cuenta microorganismos a no equivocarse ya que divide a un portaobjetos en cuadrículas microscópicas como se especificó (Estrada et al., 2002). En la Figura 3 se muestra un ejemplo gráfico de una observación de microorganismos visto en un retículo de Neubauer.

*Figura 3*

*Conteo microscópico de esporas del hongo de género trichoderma*



*Fuente:* (Thew et al., 2015)

### ***2.3.3. Determinación de la viabilidad de conidios (esporas)***

En este tipo de pruebas se observa la viabilidad de los conidios del hongo trichoderma, esto se logra primeramente con la determinación de la cantidad de conidios vivos por mL de solución a través de su germinación (solución en donde se dio la crianza del hongo en un biorreactor o placa Petri). Uno de los métodos más empleados es el denominado “diluciones seriadas” en donde la tercera dilución, desde la solución principal de crianza, se coloca en portaobjetos preparados con medio PDA. A las 24 horas se cuentan las esporas germinadas mediante un microscopio y la técnica cámara de

Neubauer (anteriormente explicada), posteriormente se calcula el porcentaje de germinación de acuerdo con el número de conidios que germinaron del total de los conidios presentes, en los 5 cuadrantes contados de la cámara de Neubauer (Caiza Sango, 2017).

### **2.3.3.1. Microscopía electrónica**

La técnica de microscopía electrónica revolucionó el conocimiento de la estructura celular dando un mayor poder de resolución en comparación a la microscopía óptica siendo posible gracias a los microscopios electrónicos (Vázquez Nin, 2000). Existen varios tipos de microscopia electrónica a continuación se citan los más importantes:

- Microscopia de luz transmitida (campo claro): es un método de microscopia electrónica que permite la observación rápida de preparados bien contrastados o teñidos. Usa una tecnología que permite el paso de luz a través del espécimen o se refleja, la iluminación no se altera por aditamentos que cambien las propiedades de la luz.
- Microscopia de luz transmitida (campo oscuro): esta tecnología permite que únicamente los rayos de luz difractados y dispersados penetren en el objetivo, mientras que los haces de luz directos no tienen influencia debido a que pasan por delante del objetivo.
- Microscopia de luz transmitida (contraste de fases): utiliza una tecnología donde el canal óptico anular, definido por el diafragma anular de fases y el anillo de fases, sirve para atenuar las partes de luz intensas y directas además de ocasionar un desplazamiento de fase constante.

### **2.3.4. Determinación de contaminantes**

Este tipo de pruebas se las realiza debido a que cualquier producción de microorganismos tienen el riesgo de contaminarse, incluso al contar en las medidas de seguridad óptimas. La prueba de determinación de contaminantes (pureza) consiente en comprobar la proporción del antagonista, en este caso del hongo de género *Trichoderma*, en relación con otros microorganismos que se han multiplicado en el medio como contaminantes, pueden ser bacterias cocos o bacillus entre otros. Esto es un proceso esencial para mejorar la producción y formulación del producto final (Oliveira, 2019).

Esta técnica se refiere a una preparación de suspensiones a partir del formulado (obtenido del cultivo en un biorreactor) específicamente del proceso de crianza del hongo de género *Trichoderma*, logrado a través de la realización de suspensiones en recipientes especiales denominados biorreactores con un sustrato líquido o sólido y una cantidad calculada del sustrato colonizado. Posteriormente se realiza varias diluciones en series, con el objetivo de implantar en cajas de Petri con APG2% (caja Petri con un medio de cultivo Agar “papa glucosado” al 2%) aquellas más diluidas.

Luego se pasa a condicionar en una estufa de cultivo. Y se espera 4 días en un periodo de incubación, para su evaluación del desarrollo de colonias que no corresponden a *Trichoderma* y al análisis de la cantidad de este hongo en la muestra, cabe recalcar que el resultado se expresa como porcentaje.

### ***2.3.5. Otras aplicaciones del hongo de género trichoderma***

Los especímenes del género *Trichoderma* son hongos filamentosos conocidos por su capacidad antagonista sobre similares ejemplares del reino Fungi, para esto utilizan varios mecanismos como la competencia por el espacio y nutrientes, la antibiosis y el microparasitismo dándole algunas posibilidades en el campo de la biología. Además de estas aplicaciones, este hongo posee diferentes funciones enfocadas en varios campos de la ciencia de la biología, enfocado principalmente en el cuidado del medio ambiente, como se observa a continuación:

#### **2.3.5.1. Degradación de contaminantes orgánicos**

Las sustancias orgánicas están hechas de carbono, hidrógeno y oxígeno, principalmente. Algunos ejemplos de compuestos orgánicos son ácidos grasos, las proteínas, los carbohidratos, así como algunos compuestos contaminantes como los hidrocarburos del petróleo, los plaguicidas, etc. (Argumedo Delira et al., 2009).

Algunas especies de hongos filamentosos incluyendo aquellas del género *Trichoderma*, pueden degradar diferentes fuentes de residuos celulósicos y lignocelulósicos. Esta capacidad enzimática, les permite a las especies de *Trichoderma* potencialmente asistir en la degradación de compuestos orgánicos contaminantes depositados en el suelo. A continuación, se presentan las interacciones de hongos del género *Trichoderma* con diferentes contaminantes de origen orgánico (Woo et al., 2014).

### **2.3.5.2. Degradación de plaguicidas, compuestos organoclorados y compuestos organofosforados**

Los plaguicidas son un foco de contaminación creados para las acciones agrícolas. Su mala administración y constancia provoca la contaminación de mantos acuíferos, suelo e incluso alimentos (Calderón Reina, 2017). Esto puede potencialmente producir variaciones en el ambiente y la salud humana, debido a que estas sustancias pueden ser bioacumuladas y biomagnificadas en la cadena alimenticia (Sánchez Salinas & Ortiz Hernández, 2018).

Varias especies del hongo de género *Trichoderma* tienen la capacidad de degradar plaguicidas, por su actividad enzimática. Esta capacidad bioquímica permite observar el potencial de aplicación de *Trichoderma* en la biorremediación de sitios contaminados teniendo con ello una relevancia ecológica (Gutierrez & Alayo, 2017).

### **2.3.5.3. Degradación de Ácido acrílico**

El ácido acrílico, el acrilato de metilo, el acrilato de etilo y el acrilato de butilo son usados principalmente en la elaboración de cubiertas termoplásticas, adhesivos, selladores para pinturas de látex (Polit et al., 2020). El movimiento en el suelo del ácido acrílico y de sus ésteres es muy alta; además, el ácido acrílico y el acrilato de metilo muestran poca biodegradabilidad en el suelo (Argumedo Delira et al., 2009). Ciertos hongos del género *Trichoderma* son capaces de degradar 2g L<sup>-1</sup> de ácido acrílico neutralizado con NaOH, al utilizarlo como única fuente de carbono durante cuatro días. Sin embargo, esta capacidad de degradación aumentó hasta 10g L<sup>-1</sup> al neutralizar al ácido acrílico con Ca(OH)<sub>2</sub>, por seis días (Arévalo, 2018).

### **2.3.5.4. Degradación de hidrocarburos del petróleo**

Los hidrocarburos policíclicos aromáticos (PAH, por sus siglas en inglés) son constituyentes del petróleo considerados agentes carcinogénicos y teratogénicos (Pamoarato et al., 2018), los cuales son susceptibles de ser biodegradados vía actividad microbiana reportaron la capacidad que tienen algunas especies de *Trichoderma* para degradar hidrocarburos saturados y aromáticos presentes en aceites combustibles (Muñoz Cuaical, 2016).

## 2.4. Visión artificial

Esta tecnología también conocida como visión por computador, se define como la ciencia de programar un computador o sistema embebido para procesar imágenes o videos pudiendo llegar al punto de entendimiento de estos (Farinango et al., 2016).

Según la Automated Imaging Association, la visión artificial puede destinarse a todas las aplicaciones industriales y no industriales en las que una combinación de hardware y software brinda un algoritmo al sistema para que este sea capaz de ejecutar sus funciones de acuerdo con la captación y procesamiento de imágenes (AIA, 1984).

### 2.4.1. Etapas de la visión artificial

Artificial Vision ejecuta en base a una representación de un entorno que le proporciona datos sobre brillo, colores, formas entre otros. Usualmente estas son imágenes estáticas, escenas tridimensionales o imágenes en movimiento, continuando con un conjunto de procesos destinados a realizar un análisis de esta representación (Marroquin et al., 2004).

En la siguiente sección se describe el proceso de este análisis desde la adquisición de la imagen mediante fotografía, hasta la ejecución de una acción con la información obtenida, las etapas son:

1. **Adquisición de la imagen:** Es un proceso por el cual se obtiene una imagen visual o videos, mediante una cámara digital que brinde fotografía de buena calidad, con el fin de almacenar la imagen para su proceso posterior pixel (Farinango et al., 2016).
2. **Preprocesamiento de imagen:** Manipula los datos de la imagen obtenida con el fin de facilitar y mejorar los resultados, esto depende de la aplicación en específico. Esto puede ser técnicas de quitar el color, realce, aumento de contraste, disminución de ruido, etc (AIA, 1984). Los principales objetivos de esta etapa son:
  - Disminuir el ruido creado por dispositivos electrónicos: se puede utilizar la tecnología Smoothing que se basa en el suavizado de la imagen que reduce los niveles de ruido.

- Mejorar la calidad de algunos sectores de la imagen, enfocándose al uso de píxeles.
  - Resaltar las características de la imagen.
  - Transformar la imagen a otro espacio de representación.
  - Conversión a escala de grises: se basa en la luminiscencia de la imagen o una ponderación de los componentes de color.
3. **Detección de bordes:** El borde de una imagen o fotografía digital son transiciones entre dos regiones de niveles de gris distintos. Este proceso puede suministrar una valiosa información sobre las fronteras de los objetos y ser utilizada para reconocer objetos o segmentar la imagen (Ordoñez & Oñate, 2016). Haciendo posible que solo se analice la parte de la imagen de mayor importancia para el sistema en su conjunto.
  4. **Segmentación:** La segmentación es un proceso que se encarga de dividir a una imagen o fotografía en objetos o partes constituyentes, por ejemplo seleccionar los píxeles en función de sus valores (RGB rojo, verde, azul). Se deduce que cuanto más precisa sea la segmentación el reconocimiento tiene mayor probabilidad de ser efectivo. Cabe recalcar que si se utiliza una imagen en blanco y negro se usarán 2 bits para estos 2 tonos, sin embargo, la escala de grises al poseer diferentes tonos de gris usa más de 2 bits.
  5. **Reconocimiento y localización:** El reconocimiento se puede definir como el proceso por el cual se brinda una etiqueta a un objeto basado en la información obtenida en anteriores etapas, se clasifican los objetos con características comunes dentro de un espacio y se restringe este espacio mediante técnicas de triangulación para que se seleccione una única imagen y esta pueda ser interpretada. Estas tareas que son realizadas por un algoritmo que realiza tareas en específico del procesamiento de la imagen.
  6. **Interpretación:** La interpretación se puede definir como el proceso por el cual el sistema embebido, en el caso del proyecto, toma las características delimitadas en el proceso de reconocimiento y se les brinda una predicción (Alvear-Puertas et al., 2017).

#### *2.4.2. Niveles de visión artificial*

Existe un conjunto de procesos en la percepción visual, como se colocó anteriormente, estos se pueden jerarquizar y brindar así diferentes niveles en esta tecnología, así se puede hallar inicialmente un bajo nivel, continuando por un nivel medio hasta un nivel de visión alto. Estos distintos niveles no tienen una definición precisa y aceptada, sin embargo, hay un gran consenso en cuanto a la partición de técnicas preliminares de procesamiento de imagen a nivel de píxel y técnicas que ejecutan reconocimiento e interpretación (Vencoel, 2020). A continuación, se coloca una mayor explicación de estos niveles de visión artificial:

- **Visión artificial de bajo nivel:** Este nivel bajo de visión artificial se puede definir como un preprocesamiento de imagen en donde se realiza filtrados de imágenes, detección de bordes y esquinas o morfología. Con estas acciones este nivel se considerarse como una visión por computador básica (Nedumaan et al., 2019).
- **Visión artificial de nivel intermedio:** Este nivel trabaja con la información recibida de la etapa de preprocesamiento de la imagen, para organizarla en forma de objetos o superficies. Posee dos elementos principales: la geometría y el movimiento. Así, algunos problemas que trata este nivel de visión por computador son la segmentación de objetos y el trazado visual de objetos.
- **Visión artificial de alto nivel:** Este nivel de visión artificial se adentra en la semántica de la imagen o fotografía, su eficiente entendimiento y se concentra en resolver problemas como el reconocimiento de objetos o el entendimiento de escenas.

### ***2.4.3. Aplicaciones de la visión artificial***

En los últimos años, los sistemas de visión artificial se consideran como aplicaciones tecnológicas de un alto enfoque en dominios particulares de varias áreas de la ciencia y tecnología humana, que tienen la peculiaridad de usar conocimiento de la tarea específica que debe resolver. La mayoría de las aplicaciones de visión artificial abarcan una gran variedad campos, que puede aumentar con la mejora en el rendimiento de estos sistemas (Farinango et al., 2016). Entre las aplicaciones actuales podemos destacar:

#### **2.4.3.1. Medicina**

En la ciencia de la medicina la visión artificial puede brindar una clasificación o visualización de una imagen de manera eficiente, con esta tecnología y sus resultados procesados se puede operar desde lugares remotos sin tener la necesidad de que un doctor especializado deba trasladarse al lugar rural o alejado.

En las aplicaciones en específico que se enfocan en el campo de la medicina varias investigaciones han demostrado resultados prometedores en diagnósticos complejos principalmente en las áreas de dermatología, radiología, oftalmología y anatomía patológica. Principalmente al hacer predicciones de visión por computador en el campo de la medicina la mayoría de los estudios han usado aprendizaje supervisado en conjuntos limitados de datos estructurados (Hamet & Tremblay, 2017).

#### **2.4.3.2. Inspección de calidad**

La inspección de calidad de manera general puede estar presente en los sectores de la verificación de dimensiones y fallas de un elemento o componente, específicamente en la inspección de calidad con tecnología de visión artificial, las cámaras de esta tecnología son capaces de monitorear la producción en serie de una fábrica, este proceso puede ser revisado mediante cualquier dispositivo móvil con acceso a la plataforma en la nube donde se guarde la información. Es recomendable que la plataforma sea local para evitar filtraciones de la empresa, esta acción se realiza mediante la tecnología de internet de las cosas IoT, donde también se da la opción de activar o desactivar máquinas dando un mejor control de la producción (Farinango et al., 2016).

#### **2.4.3.3. Seguridad basada en video vigilancia**

Los principales proyectos de seguridad basada en video vigilancia son los sistemas de reconocimiento facial de personas peligrosas buscadas por la ley, así teniendo como objetivo principal incrementar índices de seguridad y vigilancia en diversos lugares de alta circulación como aeropuertos. Es interesante recalcar que el reconocimiento facial se ha categorizado como una de las aplicaciones más estudiadas en áreas como la biometría, reconocimiento de patrones y procesamiento de imágenes (Maliza & Elena, 2017).

#### **2.4.3.4. Visión artificial en biología**

En el campo de la biología existen varias ramas en las que se puede aplicar la tecnología de visión artificial como la colaboración con la toma de datos de biodiversidad en una región al realizar identificación de especies locales de los reinos animalia, plantae,

fungi, protista y monera (Peña & Sanchez, 2020). El control de calidad en diferentes alimentos como carne, fruta o derivados donde usualmente se considera el color de estos alimentos biodegradables (Portero Donoso & Mena Mena, 2017).

#### **2.4.3.5. Visión artificial en la industria automotriz**

La visión artificial en la industria automotriz ha tenido un gran avance tecnológico y actualmente ciertos fabricantes tienen tecnología de punta, caracterizada por producción en masa con una enorme automatización, estrictos requisitos de calidad y un alto grado de sensibilidad a los costos (Tecnologiavao, 2019).

Específicamente las tecnologías de visión artificial se usan, en la industria automotriz, principalmente para la rápida e inequívoca detección de objetos atomizando y acelerando ciertos procesos de producción. A continuación, se coloca una lista de estas tecnologías:

- Detección de integridad.
- Inspeccionar superficies de los componentes con un alto nivel de precisión.
- Identificar y clasificar automáticamente productos defectuosos.

#### **2.4.3.6. Visión artificial en rescates**

La tecnología en cuestión posee diferentes aplicativos en esta área como búsqueda y localización de personas perdidas en zonas naturales mediante cámaras movilizadas por Drones (vehículos aéreos no tripulados) como el proyecto a implementarse en Canadá (Hispanaviacion, 2020) para la detección de incendios mediante las características ópticas del fuego o del humo, permitiendo realizar el monitoreo y la detección de incendios a mayor distancia, mejorando considerablemente otros sistemas para monitorear la presencia de fuego, están diseñados para trabajar en contacto con algún producto de la combustión, como la temperatura o la concentración de humo en el aire (González, 2018).

### **2.5. Software para visión artificial**

En visión artificial se utilizan varios lenguajes de programación, con apoyo de librerías o bibliotecas como: OpenCV que es una biblioteca escrita originalmente en C/C++, multiplataforma en cuanto a sistemas operativos e igualmente se puede ejecutar en diferentes lenguajes de programación como Java, Objective C, Python y C# (Bradski &

Kaehler, 2008) o la librería Sherlock que es un software que posee una interfaz gráfica para crear toda la programación, su principio de funcionamiento se basa en definir zonas de interés (ROI's) sobre las que se pueden aplicar diferentes funciones (Ivorra Martínez, 2021).

### ***2.5.1. Lenguaje de programación Python***

Según Díaz Algar y Fernández Vellón (2019) en su libro Introducción práctica a la programación con Python: el lenguaje de programación “Python” se basa en una estructura sencilla y versátil que da la posibilidad de implementar código de forma eficaz y sencilla, con su objetivo principal en la legibilidad de su código, Python es un lenguaje de programación interpretado de alto nivel, de fácil uso, pero capaz de aprovechar el poder de sistemas complejos de otros lenguajes cuando sea necesario.

En el lenguaje de programación de alto nivel Python al ejecutar diferentes instrucciones de un código escrito por el programador, se traducen por el intérprete a un lenguaje entendible por un dispositivo electrónico (lenguaje máquina), en el caso de este proyecto en una placa embebida, entonces se llega a la deducción de que Python es un lenguaje interpretado. Cada sentencia se ejecuta y brinda un resultado visible, este ayuda a depurar el entendimiento del código y verifica la validez de los resultados de la ejecución de pequeñas porciones de código de forma inminente (Algar Díaz & Fernández de Sevilla Vellón, 2019).

#### **2.5.1.1. Ventajas de Python**

- Python es un lenguaje multiplataforma y de código abierto, que le permite ejecutarse en gran variedad de sistemas operativos incluidas las distribuciones de Linux (Ortega Candel, 2018).
- Los códigos creados en el lenguaje de programación Python son más compactos que programas escritos en otras tecnologías como C, Java, etc. (Algar Díaz & Fernández de Sevilla Vellón, 2019).
- En diferentes repositorios en la nube existe amplia documentación enfocada al uso de Python y además su comunidad de usuarios es muy grande (Ortega Candel, 2018).

### ***2.5.2. Python en aprendizaje de máquinas***

Se han utilizado una amplia variedad de lenguajes de programación y entornos de desarrollo en el aprendizaje de máquinas o inteligencia artificial. El lenguaje de

programación de alto nivel Python ha tenido gran crecimiento de popularidad en la última década, dentro de la informática científica especializada en los ámbitos tecnológicos ya mencionados, esto debido principalmente a su comunidad sumamente activa y numerosa, las bibliotecas y herramientas de aprendizaje automático y aprendizaje profundo hacen que Python sea particularmente atractivo para cargas de trabajo en datos ciencia, aprendizaje automático e informática científica (Raschka et al., 2020).

Las aplicaciones de aprendizaje de máquina usualmente utilizan álgebra lineal, operaciones en matrices multidimensionales que son estructuras de datos computacionales para representar vectores, matrices y tensores de orden superior. Dado que estas operaciones a menudo se pueden paralelizar muchos núcleos de procesamiento, bibliotecas como NumPy y SciPy utilizan lenguajes externos como C/ C++, Fortran siempre que sea posible evitar subprocessos y otras limitaciones de Python. Estas bibliotecas son:

- La biblioteca NumPy es una biblioteca de matriz multidimensional con rutinas básicas de álgebra lineal.
- La biblioteca SciPy adorna Matrices NumPy con muchas primitivas importantes, desde optimizadores numéricos y procesamiento de señales a la estadística y álgebra lineal dispersa (Raschka et al., 2020).

## **2.6. Hardware para visión artificial**

El hardware básicamente visto para el desarrollo de aplicaciones de visión artificial se compone de un captador: esencialmente una cámara con buena resolución de píxeles (Vencoel, 2020) y un procesador: unidad funcional de un dispositivo electrónico encargado de la interpretación, búsqueda y ejecución de instrucciones (RAE, 2020). En varios casos, este hardware es considerado como un sistema embebido.

Un sistema embebido se lo puede considerar como un integrado e incrustado, es un sistema de computación constituido con el fin de realizar una o varias funciones dedicadas, usualmente a un sistema de computación en tiempo real. Entonces se puede afirmar que los sistemas embebidos son una composición de hardware y software con piezas mecánicas o digitales, los cuales debes cumplir una función específica.

Las aplicaciones de sistemas embebidos puedes ser: control de la cantidad de dinero que se debe por un trayecto en taxi (taxímetro), sistemas de control de acceso,

control de máquina expendedora o el sistema de control de una fotocopiadora, entre otras aplicaciones (Roman Bueno & Gonzalez Mantilla, 2017).

El sistema embebido usado en el presente proyecto se compone de una cámara digital conectada directamente a la placa de procesamiento. Estos componentes se integran en un sistema de mayor tamaño. En visión artificial, el PC es un elemento adecuado para realizar tareas generales de computación, mientras que las placas de procesamiento de un sistema de visión embebidas están diseñadas para llevar a cabo tareas específicas de una aplicación concreta. Las ventajas principales de este sistema son el tamaño compacto y liviano de sus componentes, su coste reducido y bajo consumo. A continuación, se observan ejemplos de estos sistemas embebidos:

- **Embebido IPS960-511-PoE:** Controlador integrado para aplicaciones de visión artificial (machine vision) que cumple los requisitos de diferentes plataformas y equipos de inspección automática.
- **NVIDIA Jetson Nano:** con una CPU ARM de 64 bits de cuatro núcleos, una GPU NVIDIA integrada de 128 núcleos y una memoria LPDDR4 de 4 GB. Ejecuta múltiples redes neuronales en paralelo y procesa varios sensores de alta resolución simultáneamente.
- **ASUS Tinker Board S:** procesador Rockchip RK3288 Quad-Core a 1,8 GHz, con un GPU ARM Mali-T764 (compatible con OpenGL ES1.1/2.0/3.0, OpenCL, DirectX11).
- **Arduino Mega 2560 R3:** con un microcontrolador ATmega2560, digital I/O Pins 54, PWM digital I/O Pins 14, pines de entrada analógica 16, memoria flash 256 KB, EEPROM 4 KB, velocidad de reloj 16 MHz.

## 2.7. Aprendizaje de máquina

El aprendizaje de máquina o Machine learning es un método en donde un artefacto electrónico puede crear y ejecutar predicciones o comportamientos mediante un base de datos previa (para su entrenamiento o aprendizaje) (Molnar, 2020). El algoritmo de machine learning requiere un conjunto de entrenamiento que consiste en datos de los que se conoce a que clase pertenecen y esta información permite aprender y crear un modelo (como red neuronal, árbol de decisión, lógica difusa etc.) a partir del cual poder clasificar

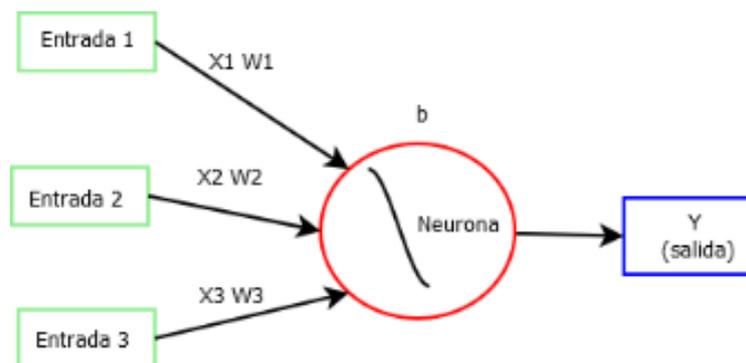
nuevos datos cuya clase es inicialmente desconocida (López de Mántaras Badia & Meseguer González, 2017).

### 2.7.1. Neurona artificial individual

El funcionamiento individual de una neurona puede observarse al conocer que trabajo cumplen cada una de sus partes (Chen et al., 2019), en la Figura 4 podemos observar un esquema del funcionamiento básico de una neurona artificial individual:

Figura 4

Esquema neurona artificial individual



Nota: **X**: valor que entrega una neurona pasada, **W**: peso de la conexión, esta tiene el fin de brindar más importancia a una neurona sobre otras o quitarle importancia a esta, **B**: bias o sesgo, valor que tiene la propia neurona con el fin de fomentar a que esta se active con mayor facilidad o con una dificultad extra, **Y**: salida del sistema.

#### 2.7.1.1. Operación para activación de neurona artificial individual

Primero se define lo que se gestiona cuando una neurona se activa, esta manda información (Y) a la siguiente neurona en la red (Martinez, 2020), este algoritmo esta supervisado por la Ecuación 1 donde se muestra la definición de gestión de una neurona activa:

$$\sum(w_n \cdot x_n) + b$$

(1)

Esta pequeña formula nos brinda dos pasos donde el primero es sumar todas las multiplicaciones entre todas las entradas por el peso de la conexión ( $w_n \cdot x_n$ ), a este resultado se le suma el valor bias o sesgo (b). Posteriormente la neurona decide si activarse o no dependiendo de que si este resultado es mayor a un número configurado en esta neurona.

### 2.7.1.2. Error cuadrático medio

Para corregir las predicciones del algoritmo es necesario el monitoreo continuo, como ya se ha mencionado, hasta tener un nivel bajo de error. Una vez hecha una operación de reconocimiento a partir de una tabla de aprendizaje, para comprobar la eficacia del sistema se utiliza el “error cuadrático medio” (Chen et al., 2019), esto se calcula en la Ecuación 2 mostrada a continuación:

$$\frac{1}{2n} \sum y(x) - a^2$$

(2)

Ecuación 2 nos da a saber que la sumatoria de  $y(x) - a^2$  elevado al cuadrado y esto multiplicado por  $1/2n$  donde:  $Y(x)$  es el valor dado por nuestra red en esa iteración,  $A(x)$  es el valor que se esperaba, esto lo decide el monitor que supervisa el algoritmo,  $N$ : número de elementos que mandamos a entrenar a la red neuronal.

### 2.7.1.3. Gradiente descendiente

Una vez conociendo el error se puede cambiar el valor de:  $b$  bias,  $w$  peso de conexión,  $x$  valor neuronal pasado. Con el fin de que este error cuadrático medio descienda. Para este proceso se utiliza el algoritmo gradiente descendiente.

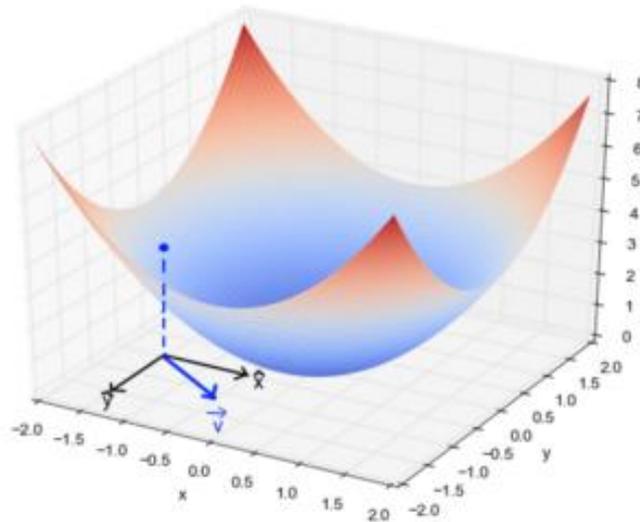
El gradiente es una generalización de la derivada, específicamente es el conjunto de todas las derivadas parciales de una función. En el caso de redes neuronales el gradiente de la función de coste es el más utilizado.

En varias técnicas de aprendizaje automático incluido redes neuronales, el aprendizaje por computador consiste en encontrar qué parámetros  $W$  (peso de conexión) minimizan el error producido. En el caso del método de gradiente descendiente se utiliza para la optimización numérica con el fin de encontrar los mejores coeficientes  $W$ .

En la Figura 5 se observa una ejemplificación del gradiente de forma gráfica, donde en este ejemplo  $Y$  representa a los parámetros  $W$  (peso de conexión) y la  $X$  representa los valores obtenidos de neuronas pasadas, su parámetro de altura nos encontramos con el “error cuadrático medio”. El objetivo es minimizar la más posible este parámetro de error cuadrático medio. NOTA: mientras más interacciones se de en la red neuronal oculta hay más probabilidades de disminuir el error cuadrático medio.

Figura 5

Gradiente en gráfico



Fuente: (Martinez, 2020)

### 2.7.2. Aprendizaje no supervisado

Los algoritmos capaces de identificar automáticamente agrupamientos y límites entre conjuntos de datos, de cualquier tipo según ciertas características, son los algoritmos de aprendizaje no supervisado. Es posible agrupar datos en clases distintas sin conocer la clase a la que pertenecen los datos contenidos en el conjunto de entrenamiento, en este caso decimos que el aprendizaje es no supervisado (López de Mántaras Badia & Meseguer González, 2017). Usualmente se busca una probabilidad que genere muestras estadísticamente similares a las muestras de datos obtenidas (Sanipatín, 2021).

### 2.7.3. Aprendizaje supervisado

El Aprendizaje Supervisado (Supervised Learning) se creó a través de redes supervisadas, estas funcionan cuando el usuario o programador brinda el primer conjunto de reglas, posteriormente son refinadas por redes neuronales mediante la ejecución a través de cientos de miles de entradas (tablas de aprendizaje), estableciendo cada vez cuan funcional es el sistema. La red tiende a mantener los cambios que mejoren del rendimiento, y hacer caso omiso de los demás (Garrido, 2020).

#### **2.7.4. Aprendizaje por refuerzo**

Los algoritmos de aprendizaje por refuerzo se pueden considerar entre los aprendizajes supervisados y los aprendizajes no supervisados. En este tipo de machine learning un agente aprende de la interacción con su medio, a partir de las consecuencias de acciones que selecciona sea con su experiencia anterior (fase de aprendizaje) o una selección aleatoria de acciones posibles en cada situación, realizando un proceso de prueba y error. En este último se utiliza el denominado agente que recibe un valor numérico llamado refuerzo (positivo o negativo), el objetivo de este es seleccionar las acciones que maximizan el refuerzo acumulado. Así, al aprendizaje por refuerzo no se lo toma como aprendizaje supervisado porque no se proporcionan ejemplos correctos de pares estado y acción, pero tampoco es completamente no supervisado debido a que el agente recibe un valor de refuerzo que guía su aprendizaje.

#### **2.7.5. Deep learning**

Se conoce como deep learning o aprendizaje profundo a un conjunto de métodos de aprendizaje automático de datos con varias capas de abstracción, donde se aplica en cascada una serie de transformaciones no lineales, así cada nivel recibe en su entrada la salida del anterior nivel y ejecuta una abstracción, aprendiendo así representaciones cada vez más complejas (López de Mántaras Badia & Meseguer González, 2017).

##### **2.7.5.1. Red neuronal artificial ANN**

Las redes neuronales artificiales (ANN) se pueden definir como un subcampo del aprendizaje automático o deep learning. El objetivo de la tecnología ANN es desarrollar sistemas de aprendizaje automático tomando como inspiración el modelo biológico del cerebro humano, de ahí se toma el nombre de neurona artificial (Malik, 2018) donde cada neurona artificial simboliza una unidad de proceso que forma parte de una entidad mayor denominada como red neuronal artificial ANN profundas (Lee et al., 2020).

##### **2.7.5.2. Red neuronal profunda DNN**

Las redes neuronales profundas usan una jerarquía de niveles en cada capa, lo que le permite aprender y clasificar el objeto deseado y sus similitudes con todos los detalles presentes en la imagen. Puede utilizar una combinación de aprendizaje supervisado y no supervisado en las diferentes capas ocultas internas, así puede reconstruir la fotografía o

imagen de entrada obteniendo características propias en cada una de ellas (Madhavan & Jones, 2017).

Los parámetros de la red comienzan con una suposición, luego de pasar los datos por toda la red, se obtiene el índice de error y la influencia de cada parámetro; así se continúa actualizando sus parámetros. Unas veces entrenada la red de este modo, es capaz de realizar su tarea con datos no etiquetados, los cuales la red desconocía (Pasquel, 2018).

### **2.7.5.3. Red neuronal recurrente RNN**

La red neuronal recurrente RNN se puede definir como un algoritmo robusto con memoria interna, esta tecnología tiene conexiones de retroalimentación entre capas o a la misma capa lo cual permite una memoria de entradas pasadas. Esto le permite al algoritmo ser capaz de recordar y aprender información importante sobre la entrada que recibió además de requerir de mayor tiempo de procesamiento. Así podemos decir que una red multicapa, como RNN, se diferencia de una red recurrente debido a sus conexiones (Ahmed & Brifcani, 2019).

Las redes neuronales recurrentes RNN poseen un extenso conjunto de arquitecturas, todas con su característica principal de retroalimentación y con un comportamiento variante en el tiempo, lo cual permite resolver problemas con no-linealidades temporales significativas, es decir, son ideales para la identificación y clasificación de patrones secuenciales que tengan diferentes probabilidades de suceder en el tiempo (Oropeza Clavel, 2007).

### **2.7.5.4. Redes neuronales convolucionales CNN**

Las CNN son el algoritmo utilizado en Aprendizaje Automático para dar la capacidad de “ver” al ordenador. La CNN es un tipo de Red Neuronal Artificial con aprendizaje supervisado que procesa sus capas imitando al córtex visual del ojo humano para identificar distintas características en las entradas que en definitiva hacen que pueda identificar objetos y “ver”. Para ello, la CNN contiene varias capas ocultas especializadas y con una jerarquía: esto quiere decir que las primeras capas pueden detectar líneas, curvas y se van especializando hasta llegar a capas más profundas que reconocen formas complejas como un rostro o la silueta de un animal (Núñez Sánchez, 2016).

## 2.8. Edge computing

Según Shi & Dustdar (2019) la gran acogida de la tecnología IoT juntamente con los servidores en la nube han asistido a establecer la necesidad de la informática de borde (Edge Computing), en la que el procesamiento de datos se produce en la parte perteneciente al borde de la red, en lugar de realizar esta acción en la nube. Edge Computing acerca los recursos informáticos y de almacenamiento a los usuarios finales y las fuentes de datos (Gedeon et al., 2019). Siendo el principal objetivo de esta tecnología el de que sus nodos en el borde de la red realicen las operaciones informáticas como el procesamiento de datos, el almacenamiento en caché, la administración de dispositivos y la protección de la privacidad, para reducir el tráfico de los dispositivos a la nube (Shi & Dustdar, 2016).

### 2.8.1. Arquitecturas de referencias de Edge Computing

Para esta temática se toma como referencia dos de las AR (arquitecturas de referencia) de Edge Computing basadas en la norma ISO/IEEE 42010:2011. Esta arquitectura de referencia fue trazada en el marco del Proyecto FAR-EDGE (Factory Automation Edge Computing Operating System Reference Implementation) con financiación del programa de investigación e innovación Horizonte 2020 de la Unión Europea (FAR-EDGE, 2017). Los ítems principales de esta arquitectura de referencia FAR-Edge son ámbitos (scopes) y niveles, a continuación, sus definiciones:

- **Los scopes:** son los elementos del ecosistema Edge Computing con referencia a la arquitectura FAR-EGDE como maquinaria, dispositivos de campo, estaciones de trabajo; sistemas SCADA (Supervisión, Control y Adquisición de Datos); MES (Sistema de Ejecución de Fabricación) y ERP (Sistema de Planeación de Recursos Empresariales).
- **Los tiers:** proporcionan información sobre los componentes de la arquitectura y su relación entre ellos (FAR-EDGE, 2017).

A continuación, una lista de las principales capas de esta arquitectura:

- **Campo o Industria:** primera capa de la arquitectura FAR-EDGE, está formada por diferentes tipos de dispositivos que pueden ser: teléfono inteligente, un PDA (asistente digital personal), hasta un sensor o un actuador, en el caso específico del proyecto una cámara digital con acople a microscopio

- **Edge:** podemos encontrar los dispositivos Edge que funcionan como puertas de enlace entre el nivel de Campo y el entorno digital. Su principal función es realizar un análisis de los datos en tiempo real. En este proyecto en específico viene a ser la placa embebida donde se procesa datos y se sube a una nube los resultados.
- **Cloud:** consiste en servidores (en la nube) que alojan el software responsable de la planificación, monitorización y gestión de los recursos. Esta capa se ocupa de la ejecución lógica de los componentes funcionales de la arquitectura.

### 2.8.2. *Edge Computing 2.0*

El Edge Computing Consortium (ECC) fue publicado en el año 2016 por varias organizaciones como: Huawei, Shenyang Institute of Automation (SIA) de la Academia China de Ciencias, Intel, ARM, iSoftStone y la Academia China de Tecnología de Información y Comunicaciones (CAICT). Esto dio como resultado que la Arquitectura de Referencia de Edge Computing (EC-RA) 2.0 fue propuesta, basándose en estándares internacionales principalmente el ISO/IEC/IEEE 42010:2011 (ISO/IEC/IEEE, 2011). Horizontalmente, sigue un modelo de capas con interfaces abierta, que se describe en el siguiente orden:

- **Servicios Inteligentes:** esta capa está basada en un marco de servicios basado en modelos. La coordinación inteligente entre el desarrollo y el despliegue de servicios se logra a través del marco de servicios de desarrollo y el marco de servicios de despliegue y operación. Estos marcos permiten interfaces coherentes de desarrollo de software e implementación y operaciones automáticas.
- **Servicios de fábrica:** define las tareas, los procesos tecnológicos, los planes de ruta y los parámetros de control de las fases de procesamiento y ensamblaje, implementando el rápido despliegue de políticas de servicio y el rápido procesamiento de múltiples tipos de productos.
- **Cómputo y conectividad:** La infraestructura de Operación, Tecnología de la Información y las Comunicaciones (OICT) se encarga de desplegar las operaciones y coordinar entre los servicios de recursos computacionales y las necesidades de la organización (Sittón-Candanedo et al., 2019).

## 2.9. Trabajos relacionados

Para abordar este proyecto se analizan algunos trabajos relacionados con enfoque en la ciencia de la biología, que permiten ampliar el panorama de la solución que se desea proponer, en este contexto se toma como referencia las siguientes investigaciones:

El trabajo desarrollado por Naranjo y Silva (2016) denominado: “Diseño e implementación de un sistema para la contabilización automática de unidades formadoras de colonias bacterianas: coliformes, mesófilos aerobios, bacilos, estafilococos y levaduras usando técnicas de visión artificial”, expone la creación de un prototipo hardware y software para obtener precisión y disminuir considerablemente el tiempo de recuento de colonias bacterianas en caja Petri en el laboratorio de Microbiología de Biotecnología en la Universidad de las Fuerzas Armadas (ESPE).

En el proyecto desarrollado por la empresa “SomData (Compañía GOHUB)” se aprecia un sistema de visión artificial que permite la detección de microorganismos en muestras de agua purificadas, con el objetivo de identificar cualquier organismo que pueda ser considerado peligroso en esta sustancia apta para el consumo humano (Pastor, 2019).

La siguiente investigación científica desarrollada por Alvarez Ramos, Nino, & Santos (2013) con título: “Automatic classification of Nosema pathogenic agents through machine vision techniques and kernel-based vector machines”, la cual muestra un método simple y directo para identificar esporas con el uso de técnicas de visión artificial y aprendizaje supervisado con el objetivo de detectar enfermedades en colonias de abejas. El método usa técnicas de segmentación para identificar las esporas que se clasifican mediante máquinas vectoriales basadas en núcleos de múltiples clases.

En el proyecto desarrollado por Yordanov & Mladenov (2018) denominado: “Humanoid robot detecting animals via neural network”, dicha investigación científica da a conocer una red neuronal con el fin de identificar seres del reino animal mediante imágenes de distintos animales para su entrenamiento. Las nuevas fotografías se obtienen por la cámara de un robot. Luego se procesa a través de una red neuronal de convolución que se implementa con la librería Tensorflow en un servidor, así el robot puede identificar y mencionar el nombre del animal.

Para finalizar, la investigación científica hecha por Militante, Gerardo, & Dionisio (2019) tuvo como objetivo diseñar el proyecto “Plant leaf detection and disease recognition using deep learning”, enfocado en brindar una solución eficaz que permita detectar múltiples enfermedades en un grupo de plantas de una variedad definida. El sistema puede detectar y reconocer varios tipos de enfermedades en las siguientes plantas: manzana, maíz, uva, papa, caña de azúcar y tomate. Posee una base de datos de entrenamiento de 35000 imágenes de hojas de plantas sanas e infectadas, se logró entrenar modelos de aprendizaje profundo para reconocer si existen enfermedades en estas plantas o la ausencia de ellas.

Los trabajos citados permiten conocer la aplicación de la inteligencia artificial dentro de la biología. Estas investigaciones demuestran que la tecnología relacionada con inteligencia artificial tiene un gran impacto dentro de este campo de estudio al permitir a los investigadores identificar especies tanto del reino animal, reino vegetal incluido el reino fungi. Además, ayuda en la detección temprana de enfermedades en plantas, animales o desastres naturales tales como incendios en zonas remotas o el reconocimiento de microorganismos nocivos en sustancias para consumo humano.

### 3. CAPÍTULO III. Diseño del sistema

El capítulo III se realiza en base a las fases de gestión de proyecto del modelo PMBOK en donde específicamente se toma los procesos de recopilación de requisitos y gestión de integración del proyecto enfocándose en los procesos de dirección y gestión. Para llevar a cabo de manera más práctica estos procesos dentro de este capítulo, se establece el proceso de diseño del sistema basado en el modelo secuencial en cascada. Haciendo uso de técnicas basadas en este modelo se lleva a cabo la recolección de requisitos que incluye la selección de hardware y software; el diseño y programación para adquisición y el procesamiento de imágenes usando el lenguaje de alto nivel Python en el hardware por seleccionar, pruebas de funcionamiento del sistema resultante y una fase final de entrega que abarca la búsqueda de errores y la aprobación del sistema.

#### 3.1. Metodología

El presente proyecto se encuentra orientado a la creación de un dispositivo electrónico haciendo uso de una placa embebida para el procesamiento, con el objetivo principal de desarrollar un sistema de identificación y conteo de esporas del hongo de género trichoderma basado en técnicas de visión artificial para contribuir al proceso de control de calidad en la crianza de este microorganismo. Se utiliza el “Modelo secuencial en Cascada” para el diseño e implementación del sistema, incluyendo la selección de hardware, software, requisitos del sistema y de usuarios (con sus respectivas características).

##### 3.1.1. Modelo en cascada

El modelo en cascada se considera como un procedimiento lineal que divide los procesos de desarrollo en fases sucesivas de proyecto, enfocado en el desarrollo de software y pudiendo acoplarse al desarrollo de hardware. Por lo cual se escogió este proceso exclusivamente para la creación y pruebas del sistema contador de esporas de hongo trichoderma con técnicas de visión artificial (Patricia et al., 2018). A continuación, se muestran los respectivos procesos:

- **Análisis:** Actividad donde principalmente se observarán con detalle los requisitos tanto de software como hardware del proyecto y un análisis de definición de requisitos en el cual los problemas complejos se dividen en tareas consecutivas.

- **Diseño:** Actividad cuyo objetivo principal es la creación de una solución que satisfaga las exigencias y estrategias colocadas en la fase de análisis. Los desarrolladores deben diseñar la arquitectura del sistema y un plan de diseño detallado, centrándose en componentes concretos.
- **Implementación:** La arquitectura del sistema se ejecuta en la fase de implementación; la cual incluye la programación del software, adquisición y manejo de hardware, la búsqueda de errores y las pruebas individuales. Cada componente se desarrolla por separado, luego de ser comprobados mediante pruebas individuales se incorporan poco a poco al producto final.
- **Pruebas:** La fase de prueba se refiere a la integración del sistema en el entorno seleccionado. Por regla general, el producto se envía en primer lugar a los usuarios finales seleccionados. El objetivo de las pruebas de aceptación de la fase de análisis es determinar si el sistema cumple con las exigencias especificadas.
- **Entrega:** Esta fase incluye la entrega y mejora tanto del software como hardware.

### 3.2. Análisis

El análisis tiene como objetivo establecer los requerimientos del sistema en cuanto al software y hardware utilizados en el diseño, en donde se incluye una placa embebida con requisitos establecidos y una cámara digital con acople a microscopio (el cual BiosebOrganics tiene inventariado antes del desarrollo de la tesis). Además, se realiza un análisis de la situación actual del sistema a partir de la información obtenida de las técnicas de investigación, en específico la encuesta aplicada, que se adjunta más adelante.

#### 3.2.1. Técnicas de investigación de recopilación de requisitos

Se utilizan diferentes técnicas de investigación para la recopilación de los requisitos del sistema con el objetivo de hacer una indagación objetiva. La primera es el cuestionario aplicado al Ing. Gabriel Curipallo con la finalidad de fortalecer conocimientos específicos acerca de los métodos de verificación de calidad en las mezclas de crianza de hongos de género trichoderma y obtener recursos utilizables que

posea la empresa. Cabe resaltar que este cuestionario es exclusivo para el personal de laboratorio de BioserbOrganics.

La segunda técnica es la revisión de trabajos de investigación relacionados, en donde es necesario localizar etapas tanto de software como hardware del sistema basado en técnicas de visión artificial, herramientas o librerías de software para cada una de las etapas y tipos de algoritmos compatibles con visión artificial.

Para finalizar, se utiliza la técnica denominada mesa de trabajo en donde se hace un compendio de los requisitos y características resultantes de las técnicas anteriores, además de la localización de requisitos adicionales necesarios para el presente proyecto.

### **3.2.1.1. Cuestionario aplicado al ingeniero encargado**

El cuestionario se aplicó al Ing. Gabriel Curipallo, encargado del proceso de crianza del hongo de género trichoderma incluida la etapa de control de calidad, con el objetivo de recopilar información referente al proceso de crianza del hongo enfocándose en la etapa de revisión de calidad de los resultados.

El cuestionario tiene un formato mixto de preguntas abiertas y cerradas, a continuación, en la Tabla 2 se observan las preguntas y en el ANEXO B se muestra el cuestionario con las respectivas respuestas.

*Tabla 2*

*Preguntas del cuestionario*

<b>Cuestionario</b>		
<b>Nro</b>	<b>Pregunta</b>	<b>Opciones</b>
1	¿Qué materiales que la empresa posee y se pueda usar en el presente proyecto?	<ul style="list-style-type: none"> <li>• Microscopio</li> <li>• Microscopio digital</li> <li>• Cámara con acople a microscopio con soporte para Linux</li> <li>• Placa embebida</li> <li>• Pantalla</li> <li>• Maus y teclado</li> <li>• Otros:</li> </ul>
2	¿Qué forma sería más cómoda de usar en el sistema?	<ul style="list-style-type: none"> <li>• Tamaño aproximado a un computador de escritorio</li> <li>• Tamaño aproximado a una laptop (computador portátil)</li> </ul>

		<ul style="list-style-type: none"> <li>• Tamaño pequeño (equiparable a un mini PC o NUC)</li> </ul>
3	¿Qué métodos de control de calidad usan?	Zona escrita
4	¿Qué técnica de conteo de esporas utilizan?	Zona escrita
5	¿Cuáles son las cualidades visibles, microscópicamente, de la espora de hongo trichoderma?	Zona escrita
6	¿Cómo diferenciar las esporas de hongo trichoderma en relación con posibles microorganismos invasores?	Zona escrita
7	¿Qué microorganismos invasores se han detectado?	Zona escrita
8	¿Qué procedimiento se realiza una vez hecho el conteo de esporas de hongo trichoderma?	Zona escrita
9	¿Qué procedimiento realizan al encontrar en el control de calidad cualquier cantidad de microorganismos invasores?	Zona escrita

---

### 3.2.1.2. *Revisión de trabajos relacionados*

La revisión de investigaciones relacionadas con el tema del proyecto de grado tiene el fin de establecer requerimientos técnicos tecnológicos funcionales para el diseño y la implementación del sistema contador de esporas de hongos del género trichoderma mediante técnicas de visión artificial. De cada uno de los trabajos revisados se tomó una cita mediante ficha bibliográfica, como se muestra en el ANEXO C, además de esto, se extrajo información relevante específicamente en la sección de elección de requisitos. A continuación, en la Tabla 3, se observa una ficha bibliográfica básica de los cuatro proyectos de esta sección, en el ANEXO C se encuentra la ficha bibliográfica completa.

Tabla 3

Ajuste ficha bibliográfica 1

<b>Ficha bibliográfica 1</b>			
Autor:	Naranjo y Silva	Editorial:	Universidad de las Fuerzas Armadas
Título:	“Diseño e implementación de un sistema para la contabilización automática de unidades formadoras de colonias bacterianas: coliformes, mesófilos aerobios, bacilos, estafilococos y levaduras usando técnicas de visión artificial”		
Año:	2016	Ciudad:	Quito
<b>Ficha bibliográfica 2</b>			
Autor:	Álvarez Ramos, Nino, & Santos	Editorial:	IEEE
Título:	“Automatic classification of Nosema pathogenic agents through machine visión techniques and kernel-based vector machines”		
Año:	2013	Ciudad:	Quito
<b>Ficha bibliográfica 3</b>			
Autor:	Yordanov & Mladenov	Editorial:	IEEE
Título:	“Humanoid robot detecting animals via neural network”		
Año:	2013	Ciudad:	Belgrade
<b>Ficha bibliográfica 4</b>			
Autor:	Militante, Gerardo, & Dionisio	Editorial:	IEEE
Título:	: “Plant leaf detection and disease recognition using deep learning”		
Año:	2019	Ubicación:	Euroasia

### 3.2.1.3. Mesa de trabajo

El objetivo de la mesa de trabajo es realizar una reunión técnica tecnológica para constatar todos los elementos necesarios, en cuanto a tecnología del sistema y reunir los requisitos obtenidos con las anteriores técnicas de investigación (encuesta y revisión de trabajos relacionados), poner en revisión otros requisitos necesarios para el sistema y brindar posibles candidatos a utilizarse en cada uno de los ítems colocados. Así, en el

ANEXO D se analiza mediante el uso de mapas conceptuales el trabajo en equipo llevado a cabo por el estudiante responsable del trabajo de grado y su director.

### **3.3. Análisis de las técnicas de recolección de información**

En el punto inicial existe un enfoque en la recopilación de información referente al proceso de crianza del hongo de género trichoderma orientado a la etapa de revisión de calidad de los resultados. Además, de un objetivo secundario que consiste en observar la situación actual de la empresa, en este ámbito se puede aseverar:

El presente trabajo de investigación se realiza en un laboratorio especializado en la crianza de diferentes tipos de hongos y bacterias aplicados a mejorar diferentes ámbitos de la agricultura como el control de plagas y tratamiento de suelos. Este laboratorio produce mezclas con una alta concentración de hongos de género trichoderma para el control de plagas específicas. Esta instalación tendrá su nueva ubicación en la ciudad de Ibarra, en la parroquia El Sagrario corresponde a la segunda cede de la empresa BiosebOrganic. Así, con la información verídica y respaldada procedente del cuestionario aplicado, se identifica claramente el instrumental que se podría usar y los aspectos técnicos biológicos de importancia para el proyecto.

#### **3.3.1. Análisis del cuestionario**

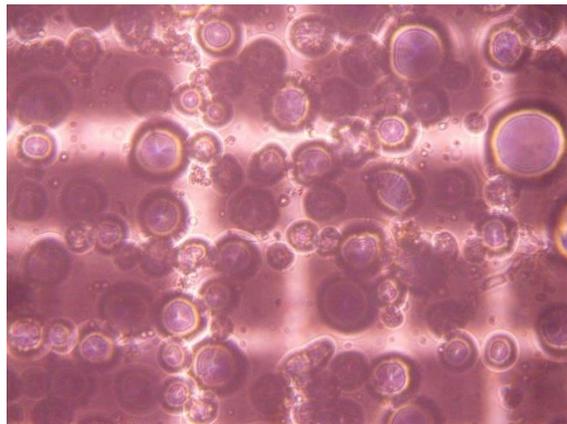
- En la primera pregunta se puede observar que la empresa tiene y está dispuesta a usar un microscopio (con la capacidad para observar microorganismos tan pequeños como son las esporas de hongo trichoderma), además de una pantalla, ratón y teclado para complementar a la placa embebida.
- En la segunda pregunta se pide que el hardware final del sistema sea pequeño y de fácil uso.
- En la tercera pregunta muestra que se usan dos métodos para el control de calidad de la crianza del hongo de género trichoderma los cuales son: determinación de la creación de cónidos o esporas y la determinación de contaminantes (microorganismos invasores).
- En la pregunta cuatro se brinda la información del método de conteo de esporas de hongo trichoderma, la cual se basa en un conteo manual con un contador electrónico.

- En la pregunta cinco se especifica la forma característica de la spora del hongo trichoderma, la cual es ovalada y con un tamaño mayor en relación con posibles microorganismos contaminantes.
- En la pregunta sexta se obtiene información de cómo diferenciar microorganismos invasores a las esporas de hongo trichoderma, la cual consiste en su forma diferente y tamaño menor.
- En la séptima pregunta se especifica los microorganismos invasores más comúnmente encontrados (bacterias bacillus y cocos), también incluye información de su forma específica: las bacterias bacillus tienen una forma más alargada en relación a las esporas del hongo trichoderma y las bacterias cocos una forma esférica.
- La novena pregunta brinda información del proceso que se realiza al encontrar un contaminante, el cual es descartar completamente la mezcla de crianza del biorreactor.
- En la pregunta número diez del cuestionario se muestra el tiempo promedio que se necesita para contar esporas de hongo trichoderma en una sola muestra.
- En la última pregunta obtenemos información de un estimado del error porcentual que puede existir al aplicar el método de conteo manual de esporas.

Cabe resaltar que se usa una disolución de 1ml de compuesto de esporas de hongo trichoderma en 10ml de líquido estéril, usualmente se utiliza una disolución en 100ml de líquido estéril, en la Figura 6 observamos una imagen del compuesto sin disolución donde se deduce que un conteo es imposible debido a que el cubreobjetos aplasta la gran cantidad de esporas.

*Figura 6*

*Compuesto de esporas trichoderma sin disolución*



### **3.3.2. *Análisis de la revisión de trabajos relacionados***

- En el trabajo relacionado 1 se sintetiza una gama de requisitos utilizados en este proyecto como son: Labview para desarrollo de visión artificial, la herramienta ROI para tratamiento preprocesamiento de imagen; además en su hardware se aprecia una cámara fotográfica Nikon modelo D90 con su respectivo software para adquirir imagen en un ordenador.
- En el trabajo relacionado 2 se observa, en sus elementos hardware, un microscopio digital para adquisición de imágenes de microorganismos y en sus elementos software se mencionan los procesos de segmentación y extracción de características de imagen, sin embargo, no se coloca que software se utilizó para estos procesos.
- En el trabajo relacionado 3 se especifica el uso de una red neuronal para la clasificación de perros y gatos en fotografías utilizando el software Tensor Flow en lenguaje de alto nivel Python.
- En el trabajo relacionado 4 se usa un repositorio existente de plantas y vegetales (Plant Village), programada en el lenguaje de programación Python, en el preprocesamiento de imagen se aprecia que cambian la escala de color, redimensionan y suavizan la imagen; para el procesamiento se utiliza una red neuronal convolucional en el lenguaje ya especificado.

### **3.3.3. *Análisis de la mesa de trabajo***

- En cuanto a adquisición de imagen tenemos como requisitos: compatibilidad con Linux, capacidad de acople a microscopio o logro para tomar fotos de microorganismos (microscopio digital), resolución mínima 5Mpx y costo no superior a \$350. Además de posibles dispositivos para estos requisitos como son: cámara fotográfica, cámara fotográfica con acople a microscopio, microscopio electrónico y Smart Phone.
- Referente a la Placa embebida los requisitos son los siguientes: compatibilidad con Linux, buena capacidad de memoria RAM, consumo de energía eléctrica reducido, buena capacidad en el rendimiento de su CPU, capacidad de contener una tarjeta de memoria MicroSD. Además de posibles dispositivos para estos requisitos como son las placas embebidas: ASUS Tinker Board, Nvidia Jetson Nano, Arduino Mega 2560, RaspBerry pi 4.

- En tratamiento de imagen se colocan varios procesos como suavizado de imagen, segmentación de imagen y escala blanco y negro de imagen, esto se realiza con librerías compatibles con el lenguaje de programación Python que son: Numpy, Opencv y LabelImg.
- En la sección de lenguaje de programación se escogió el lenguaje de programación de alto nivel Python, debido a que cumple con todos los requisitos planteados como son: compatibilidad con Linux, lenguaje de alto nivel, cantidad de código reducido y cuenta con librerías de visión artificial.
- En el requisito de técnicas de visión artificial empleando aprendizaje profundo se tiene varias opciones como son: Redes neuronales artificial (ANN), Redes neuronales convolucionales (CNN), Redes neuronales recurrentes (RNN).

### 3.4. Requerimientos del sistema

Como siguiente fase del análisis del sistema de conteo de esporas de hongo trichoderma mediante técnicas de visión artificial, se encuentra la etapa de definición de requerimientos, donde se emplea parámetros del modelo secuencial en cascada para desarrollo de software. Este ítem tiene como fin el establecer los elementos funcionales para lograr una visión artificial usando técnicas de aprendizaje profundo (visión artificial). A continuación, se observa la Tabla 4 donde se especifica los acrónimos a utilizarse en la selección de requerimientos del sistema y en la Tabla 5 se muestra las especificaciones a usarse en posteriores listados detallados de requisitos donde se describe el nivel de importancia que tiene cada requerimiento del sistema:

*Tabla 4:*

*Descripción de acrónimos*

<b>Acrónimo</b>	<b>Descripción</b>
<b>SsRS</b>	Requerimientos del stakeholder
<b>SiRS</b>	Requerimientos del sistema
<b>SHRS</b>	Requerimientos de Software y Hardware

#### 3.4.1. Beneficiarios

Los beneficiarios del sistema se clasifican en directos e indirectos. En la Tabla 5 se presenta una lista de personas e instituciones beneficiarias del presente proyecto:

Tabla 5:

Tabla de beneficiarios

<b>BENEFICIARIOS</b>	
Beneficiarios directos	<ul style="list-style-type: none"><li>• Autor del presente trabajo de titulación: Esteban Noboa</li><li>• Trabajadores encargados de la crianza de hongos trichoderma para control integral de plagas en la empresa BiosebOrganics.</li><li>• Personas encargadas de la línea de producción de productos relacionados con el hongo trichoderma y los administradores de la empresa Bioseb Organics.</li><li>• Empresa BiosebOrganics</li><li>• Director: Ing. Fabián Cuzme</li><li>• Asesor 1: Ing. Luis Suarez</li><li>• Asesor 2: Ing. Pamela Godoy</li></ul>
Beneficiarios indirectos	<ul style="list-style-type: none"><li>• Comunidad de estudiantes de la Universidad Técnica del Norte que tengan un interés en la temática de visión artificial.</li><li>• Docentes de la Universidad Técnica del Norte</li><li>• Comunidad científica en general.</li><li>• Clientes consumidores de productos en base a hongo trichoderma de la empresa BiosebOrganics.</li></ul>

En la Tabla 6 se muestra la descripción de etiquetas que se usan en las próximas tablas de comparación de requerimientos, en específico las importancias alta, media y baja decididas en el ANEXO D.

Tabla 6:

Especificaciones para tabla de requerimientos

<b>ESPECIF.</b>	<b>DESCRIPCIÓN</b>
Alta	Requerimiento crítico que debe ser considerado en el desarrollo del sistema.
Media	Requerimientos que pueden ser omitidos solo en condiciones de fuerza mayor.

Baja	Requerimientos que se pueden omitir, este requerimiento no genera impacto significativo en el sistema.
------	--

### 3.4.1.1. Requerimientos de stakeholders

Los stakeholders son personas o entidades que participan directa o indirectamente en el desarrollo del trabajo en este caso del sistema de conteo de esporas de hongo trichoderma con técnicas de visión artificial. A continuación, se muestran los requerimientos brindados por el Ing. Gabriel Curripallo obtenidos del cuestionario que se encuentra en el ANEXO B, la revisión bibliográfica identificada en el ANEXO C y la mesa de trabajo en el ANEXO D. Se obtuvo mediante la encuesta una lista de necesidades en calidad de representante de personal encargado del control de calidad de la crianza del hongo trichoderma son considerados como usuarios directos. En la Tabla 7 a continuación se puede ver una lista de todos los requisitos del sistema de conteo de esporas del hongo trichoderma mediante técnicas de visión artificial.

Tabla 7

Tabla de requisitos de usuario

		Prioridad		
		Alta	Media	Baja
<b>Requerimientos de stakeholders (usuarios)</b>				
<b>SsRS1</b>	• Cámara con acople a lente de microscopio electrónico	X		
<b>SsRS2</b>	• Dispositivo pequeño y manipulable, con protección		X	
<b>SsRS3</b>	• Identificación de cualidades visibles a través de microscopio del microorganismo tipo espora del hongo de género trichoderma	X		
<b>SsRS4</b>	• Identificación de objetos que no coincidan con las cualidades de la espora del hongo trichoderma			X
<b>SsRS5</b>	• Agilización del tiempo de conteo de esporas del hongo de género trichoderma	X		
<b>Requerimientos operacionales</b>				
<b>SsRS6</b>	• Método de adquisición de imagen	X		
<b>SsRS7</b>	• Técnica de visión artificial	X		
<b>SsRS8</b>	• Almacenamiento en base de datos			X

SsRS9	• Operación permanente	X
-------	------------------------	---

### Requerimientos iniciales del sistema

En este apartado se definen ítems, Tabla 8, siendo estos hallados en el análisis de trabajos relacionados mediante la técnica de recopilación de requisitos fichas bibliográficas, ANEXO C, estos autores son beneficiarios indirectos al pertenecer a la comunidad científica en general. Para finalizar se adjuntan los requisitos adicionales obtenidos de la mesa de trabajo, ANEXO D, realizada entre el autor del proyecto y el director siendo ambos beneficiarios directos.

Tabla 8

Tabla de requisitos iniciales del sistema

<b>TABLA DE REQUERIMIENTOS DEL SISTEMA</b>		
		<b>Prioridad</b>
		<b>Alta    Media    Baja</b>
<b>Requerimientos funcionales</b>		
<b>SiRS1</b>	• Identificar la forma de las esporas de hongo trichoderma visibles a través de un microscopio	X
<b>SiRS2</b>	• Identificar figuras no coincidentes con esporas de hongo trichoderma	X
<b>SiRS3</b>	• Contar la cantidad de esporas en la imagen obtenida	X
<b>SiRS4</b>	• Mostrar resultados de dicho conteo	X
<b>Requerimientos de uso</b>		
<b>SiRS5</b>	• Simplicidad del manejo del sistema	X
<b>SiRS6</b>	• Manual de usuario y configuración	X
<b>Requerimientos de performance</b>		
<b>SiRS7</b>	• Menor consumo de energía	X
<b>SiRS8</b>	• Soporte de un lenguaje de programación de alto nivel	X
<b>SiRS9</b>	• Técnica de visión artificial basada en aprendizaje profundo	X
<b>SiRS10</b>	• Almacenamiento de resultados en nube	X

<b>SiRS11</b>	• Cambio de escala de color en imágenes	X
<b>SHRS12</b>	• Librerías para segmentación de imágenes	X
<b>Requerimientos de interfaces</b>		
<b>SiRS13</b>	• Comunicación cámara hacia placa embebida a través de puerto USB	X
<b>SiRS14</b>	• Puertos red alámbrica ethernet	X
<b>Requerimientos físicos</b>		
<b>SiRS15</b>	• Tamaño reducido	X
<b>SiRS16</b>	• Protección frente a manipulación de trabajadores	X
<b>SiRS17</b>	• Protección hacia un ambiente aclimatado de aproximadamente 24°C a 26°C.	X

### 3.4.1.2. Requerimientos de hardware y software del sistema

Se detalla todo el software y hardware de sistema necesario siendo estos libres o pagados. Se toma estos datos del ANEXO D “Mesa de trabajo” donde se adjuntan datos tomados de los ANEXOS B y C que además se complementa mediante la participación del autor y el director del trabajo de grado. La Tabla 9 indica las cualidades en estos dos ámbitos:

Tabla 9

Tabla de requisitos funcionales del sistema

<b>TABLA DE REQUERIMIENTOS FUNCIONALES</b>		
		<b>Prioridad</b>
		<b>Alta    Media    Baja</b>
<b>Requerimientos lógicos</b>		
<b>SHRS1</b>	• Compatibilidad con módulos de tecnología ethernet	X
<b>SHRS2</b>	• Compatibilidad con sistemas Linux	X
<b>SHRS3</b>	• Compatibilidad con lenguaje de programación de alto nivel e interpretados	X
<b>Requerimientos de diseño</b>		

<b>SHRS4</b>	• Hardware y Software asequibles	X	
<b>SHRS5</b>	• Hardware y Software compatibles entre si	X	
<b>Requerimientos de hardware</b>			
<b>SHRS6</b>	• Placa embebida	X	
<b>SHRS7</b>	• Cámara digital con acople a microscopio, especializada en captura de imagen a través de este artefacto	X	
<b>SHRS9</b>	• Fuente alimentación de poco consumo con la capacidad de alimentar al sistema	X	
<b>SHRS10</b>	• Conexión a internet con tecnología ethernet		X
<b>SHRS11</b>	• Memoria RAM entre 2Gb a 4Gb		X
<b>SHRS12</b>	• Rendimiento CPU mayor o igual a 13,5 Gflop		X
<b>SHRS13</b>	• Ranura para tarjeta microSD		X
<b>SHRS14</b>	• Resolución de cámara de 5Mpx en adelante	X	
<b>Requerimientos de software</b>			
<b>SHRS15</b>	• Compatibilidad con librerías de visión artificial, manipulación de imágenes y aprendizaje profundo	X	
<b>SHRS16</b>	• Licencia de libre uso		X
<b>SHRS17</b>	• Compatibilidad con sistemas operativos Linux	X	
<b>SHRS18</b>	• Aplicación para adquisición de imagen compatible con Linux y cámara (especificada anteriormente)		X
<b>SHRS19</b>	• Lenguaje de programación de alto nivel	X	
<b>SHRS20</b>	• Programa para tratamiento de imagen	X	
<b>SHRS21</b>	• Técnica de visión artificial mediante aprendizaje profundo.	X	
<b>SHRS22</b>	• IDE para programación		X
<b>SHRS23</b>	• Procesamiento de imagen	X	
<b>SHRS24</b>	• Parámetros compartidos		X
<b>SHRS25</b>	• Relación espacial	X	
<b>SHRS26</b>	• Código abierto	X	

<b>SHRS27</b>	• Compatibilidad con Python	X
<b>SHRS28</b>	• Resultados subidos en servidor publicado en internet y de fácil acceso	X
<b>SHRS29</b>	• Subida de datos mediante técnica HTTP	X
<b>SHRS30</b>	• Especializado con método de microscopio de luz transmitida (campo claro)	X
<b>SHRS31</b>	• GPU para procesamiento de técnicas de visión artificial	X

---

### 3.5. Recursos

En esta sección se listan los recursos que se tiene a disposición y posteriormente se realiza la selección de cada recurso dependiendo de la importancia de los requisitos listados en la actividad anterior y seleccionados en el ANEXO D.

Los recursos destinados al proyecto son de ámbito económico, humano y tecnológico donde se encuentran tanto software como hardware. A continuación, en la Tabla 10 se muestra los recursos humanos disponibles:

*Tabla 10*

*Recursos humanos*

<b>Función</b>	<b>Nombre</b>
Tutor	Ing. Fabián Cuzme
Desarrollador	Sr. Esteban Noboa
Ingeniero a cargo del laboratorio de crianza de hongos	Ing. Gabriel Curipallo

Esta sección se realiza con el fin de establecer requerimientos técnicos tecnológicos funcionales para el diseño y la implementación del sistema contador de esporas de hongos del género trichoderma mediante técnicas de visión artificial. Además de seleccionar el más adecuado mediante tablas de comparación realizadas en base a las técnicas de recopilación de requisitos y requerimientos de usuarios ya analizados:

### 3.5.1. Selección de hardware

Esta sección permite la elección del hardware específico mediante una ponderación de requerimientos de usuarios y requisitos técnicos. Seguidamente se observa la selección de la mejor placa embebida, cámara para la recolección de imágenes y la protección externa del sistema.

#### 3.5.1.1. Placa embebida

Se requiere una placa embebida capaz de resistir la cantidad de procesamiento para los procesos de visión artificial. Por esta razón es necesario adquirir un dispositivo de esas características, además debe tener un tamaño pequeño según la petición del usuario.

Existen diferentes opciones, a continuación, se citan diferentes fabricantes como son: Nvidia, RaspBerry, Arduino, Asus. En la Tabla 11 se observa una comparación entre productos de estas marcas que cumplan con lo especificado anteriormente.

Tabla 11:

Cuadro comparativo placas embebidas

Hardw.	REQUISITOS										Val.
	SHRS	SHRS	SHRS	SHRS	SHRS	SHRS	SHRS	SiRS	SiRS	SHRS	
	2	3	6	10	11	12	13	13	15	31	
Nvidia Jetson Nano	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	10
Raspberrry PI 4	SI	SI	SI	SI	SI	SI	SI	SI	SI	NO	9
Arduino Mega 2560 R3	NO	SI	SI	SI	NO	SI	NO	NO	SI	NO	6
ASUS Tinker Board S	SI	SI	SI	SI	SI	SI	NO	SI	SI	SI	9

De la Tabla 11 se deduce que la placa embebida más adecuada en relación con el proyecto es: Nvidia Jetson Nano de 4gb de RAM debido a su compatibilidad con sistema Linux, incluido un sistema Ubuntu modificado por Nvidia para aprovechar el dispositivo

(NVIDIA Jetpack); su cantidad de memoria RAM, un consumo de energía bajo (5W) y el rendimiento de su CPU cuenta con 472 GFLOP.

### 3.5.1.2. Cámara con acople a microscopio

Se requiere una cámara de una resolución igual o mayor a la mínima recomendada para aplicaciones de visión artificial, esta debe permitir acoplar al lente de un microscopio debido a que los organismos que se desea contabilizar son microscópicos (esporas de hongo trichoderma).

Tabla 12:

Cuadro comparativo cámaras digitales

Cámara	REQUISITOS							Precio	VALOR
	SsRS 2	SiRS 13	SiRS 17	SHRS 2	SHRS 7	SHRS 15	SHRS 30		
Swift Cámara Digital para Microscopios Swiftcam	SI	SI	SI	NO	SI	SI	NO	\$115	5
Cámara para microscopios, lente de reducción	SI	SI	SI	SI	SI	SI	SI	\$265	7
Omax cámara USB Microscopio digital	SI	SI	SI	SI	SI	SI	SI	\$300	8

De la Tabla 12 se deduce que la cámara Omax USB Microscopio digital especializada para su uso en microscopios es la mejor opción debido a su compatibilidad con los sistemas operativos Linux, su forma que permite un acople a lente de microscopio, su resolución que cabe dentro del lumbral colocado en los requisitos para operaciones de visión artificial, su protección contra lo especificado en los requisitos de hardware, además de su compatibilidad con la técnica de microscopio de campo abierto y como adicional su precio se encuentra dentro del presupuesto aceptado.

### 3.5.1.3. Protección contra manipulación y clima artificial

Se requiere una caja para almacenar y proteger la placa embebida, dejar sus conectores libres y permitir su transporte. Esto se toma gracias a los requisitos de hardware referentes a protección.

Tabla 13

Cuadro comparativo protección externa

Hardware	REQUISITOS			Val.
	SiRS 15	SiRS 16	SiRS 17	
Protección tipo caja hecha en impresora 3D	SI	SI	NO	2
Protección tipo caja hecha en impresora 3D con espacio para ventilador	SI	SI	SI	3
Cajas universales UCS para sistemas embebidos	SI	SI	SI	3
Protección tipo caja realizada con láminas plásticas	SI	SI	NO	2
Protección tipo caja realizada con láminas plásticas con espacio para ventilador	SI	SI	SI	3

De la Tabla 13 se deduce que la protección adecuada es con láminas de plástico con un compartimento para ventilador debido a que ofrece protección contra manipulación y temperatura, debido a la capacidad que esta opción brinda de diseñar de manera precisa las dimensiones y puertos de la placa embebida, evitando así movimiento brusco al momento de su manipulación. Procedemos a observar sus especificaciones:

Siendo el objetivo principal el de preservar la placa embebida contra temperatura y manipulación que recibirá en el laboratorio de la empresa Bioseb Organics (su ambiente de funcionamiento). Esta fue impresa en 3 dimensiones con el material Poliacido Láctico, utilizado comúnmente en impresoras 3D debido a su buena adhesión entre capas y eficiente acabo superficial. Las medidas de la placa Nvidia jetson nano son:

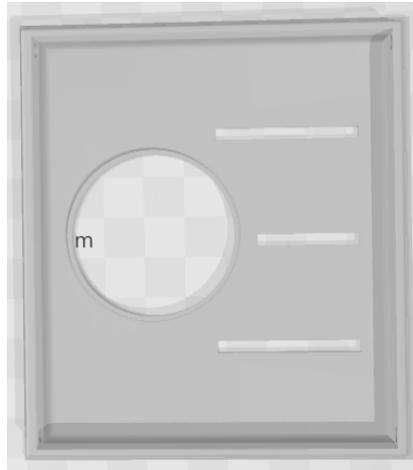
- Largo: 10cm
- Ancho: 8cm
- Alto: 3cm

Debido a que la función principal de la estructura es proteger contra manipulación y temperatura se busca un diseño que cubra componentes principales del embebido y permita la implementación de un ventilador tipo FAN pequeño sobre su disipador. Igualmente se agregan protecciones a sus puertos mediante elementos separados.

Para la creación del compartimiento para el ventilador se necesita un espacio libre en la parte superior del disipador de la placa embebida en forma cuadrada de (2.5x2.5) cm, para facilitar la ventilación se agregan hoyos rectangulares. Todo esto se realiza en la pieza superior de la protección como se observa en la Figura 7.

*Figura 7*

*Parte superior del case de protección*



Para la parte inferior se necesita un cubo hueco con espacios para los respectivos puertos de la placa embebida Nvidia Jetson nano los cuales son: puerto de carga, 2 entras HDMI de diferente tamaño, 4 puertos USB, puerto RJ45 hembra keystone y entrada USB mini. Cabe recalcar que se agregan agujeros en las esquinas superiores para realizar el acople con la pieza superior (antes vista). Esto se observa en la Figura 8.

*Figura 8*

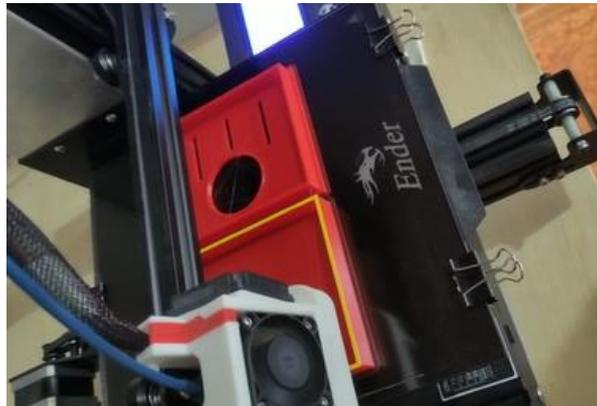
*Elemento de case inferior*



La Figura 9 muestra el proceso que realiza la impresora 3D al imprimir el diseño para la caja de protección o case antes mostrado.

Figura 9

Proceso de impresión en 3D



### 3.5.2. Selección de software

Esta sección permite la elección del software específico mediante una ponderación de requerimientos técnicos. Seguidamente se observa la selección del lenguaje de programación, el tipo de algoritmo de aprendizaje profundo a usar en las técnicas de visión artificial y el sistema que permitirá subir la información a una nube IoT.

#### 3.5.2.1. Lenguaje de programación de alto nivel Python

Se selecciona el lenguaje de programación Python debido a que su cantidad de código es menor en comparación a otros lenguajes de programación de alto nivel como JavaScript. Python es sumamente utilizado en campos de visión artificial, sus bibliotecas como Numpy, OpenCV, Scioy y Pybrain se utilizan en el aprendizaje automático y la posibilidad de usar diferentes IDE para comprobación de errores en el código. En la Tabla 14 se muestra la comparación de estos lenguajes de programación.

Tabla 14

Cuadro comparativo lenguajes de programación

Lenguaje de programación	REQUISITOS						VALOR
	SHRS 12	SHRS 15	SHRS 16	SHRS 18	SHRS 19	SHRS 22	
C++	SI	BÁSICAS	SI	SI	NO	SI	3,5
Python	SI	SI	SI	SI	SI	SI	6

Java Script	SI	SI	SI	SI	SI	SI	6
Java	SI	SI	SI	SI	NO	SI	5

### 3.5.2.2. Tipo de aprendizaje profundo (Deep Learning)

Se requiere un algoritmo de aprendizaje profundo para su implementación en el reconocimiento y conteo de esporas de hongo trichoderma mediante técnicas de visión artificial. En la Tabla 15 se compran los algoritmos de Redes neuronales artificial (ANN), redes neuronales convolucionales (CNN), redes neuronales recurrentes (RNN).

Tabla 15

Cuadro comparativo tipos de algoritmo de aprendizaje profundo

Algoritmo aprendizaje profundo	REQUISITOS			
	SHRS 23	SHRS 24	SHRS 25	VALOR
Redes neuronales artificial (ANN)	SI	NO	NO	1/3
Redes neuronales convolucionales (CNN)	SI	SI	SI	3/3
Redes neuronales recurrentes (RNN).	SI	SI	NO	2/3

De la tabla 15 se deduce que las CNN (Redes neuronales convolucionales) son la mejor opción debido a su compatibilidad para el tratado de imágenes, característica esencial para la tecnología de visión artificial, además de tener las características de parámetros compartidos, relación espacial y gradiente de desaparición.

### 3.5.2.3. Plataforma en nube compatible con Edge Computing

Se requiere una plataforma de IoT con tecnología Edge Computing para su implementación en el reconocimiento y conteo de esporas de hongo trichoderma mediante técnicas de visión artificial. En la Tabla 16 se comparan diferentes plataformas de IoT.

Tabla 16

Listado recursos de plataformas Edge Computing

Plataforma IoT	REQUISITOS				VALOR
	SHRS 26	SHRS 27	SHRS 28	SHRS 30	
UBIDOTS	NO	SI	SI	SI	3/4
THINGSPEAK	SI	SI	SI	SI	4/4
THINGWORKX	NO	SI	SI	SI	3/4

Se seleccionó la plataforma IoT ThingSpeak debido a que esta tiene un uso libre hasta un máximo de 3 millones de datos anuales, además es compatible con el lenguaje de programación Python, sus resultados son de fácil acceso y utiliza el protocolo HTTPs para subir datos a nube IoT.

### 3.6. Diseño del sistema

En esta sección se detalla el diseño del sistema, teniendo en cuenta las tablas de comparación que se hallan en el ítem 3.5. Los requerimientos de los usuarios tomados principalmente de la encuesta (ANEXO B), elementos tomados de revisión de trabajos relacionados (ANEXO C) y requisitos adicionales resultantes de la mesa de trabajo (ANEXO D). Cabe destacar que haciendo uso de diagramas de bloque y flujo se muestra el funcionamiento del sistema al brindar información acerca de los procesos llevados a cabo en el presente proyecto.

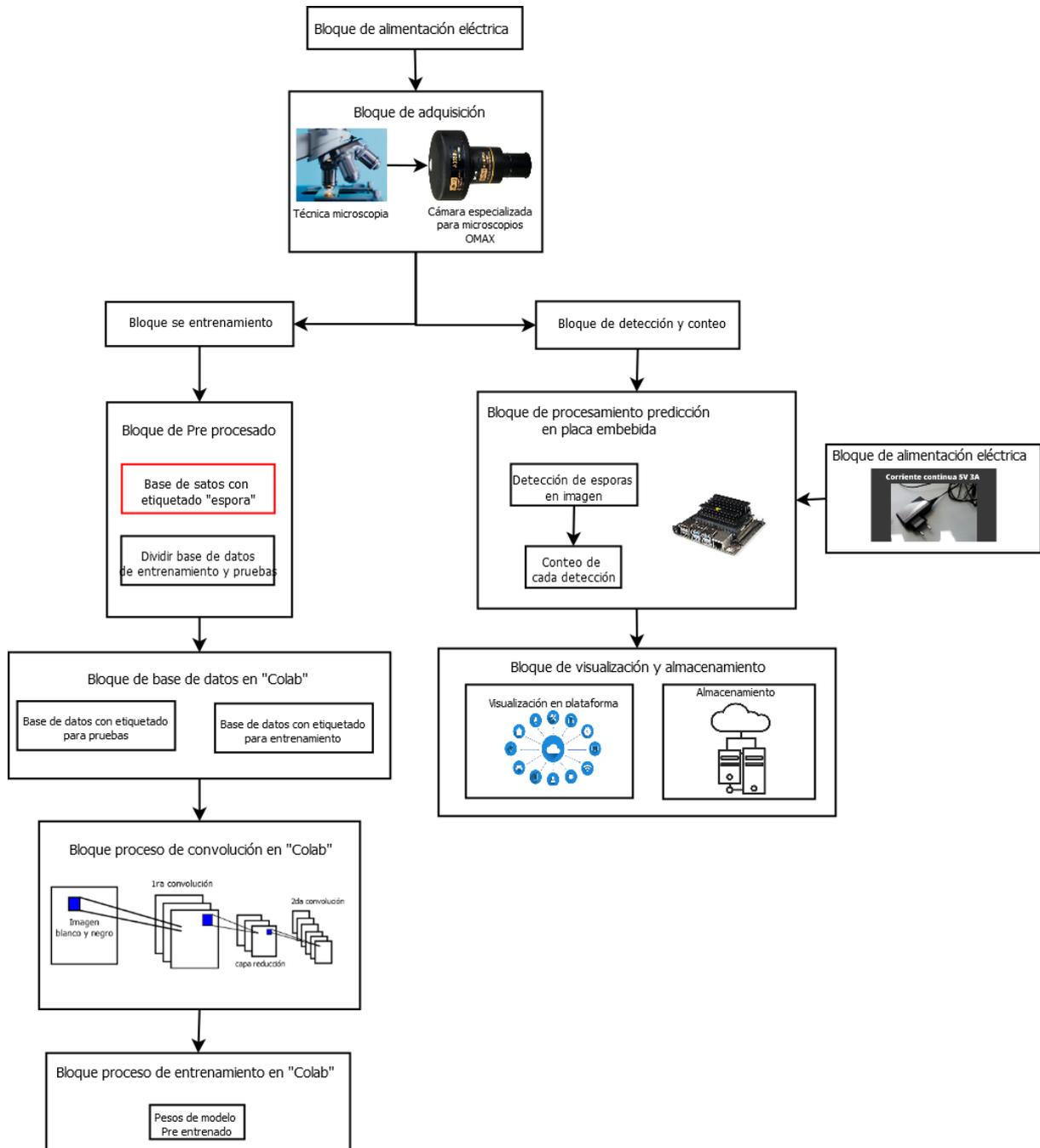
#### 3.6.1. Diagrama y flujograma del sistema

El sistema de detección y conteo de esporas de hongo trichoderma se divide en dos procesos realizados en diferentes plataformas las cuales son: “Google Colab” encargado de las acciones de entrenamiento y la placa embebida “Nvidia Jetson Nano” que se ocupa del proceso de detección y conteo en base a los resultados del proceso de entrenamiento. Cabe recalcar que el proceso de entrenamiento se lo realiza una sola vez y se utiliza los archivos resultantes para realizar repetidamente la detección y conteo de esporas en la placa embebida.

La Figura 10 representa el diagrama general del sistema, en donde se describen básicamente los procesos principales del sistema ya mencionados, posteriormente se profundizará a detalle cada uno de estos juntos con sus respectivos subprocesos.

Figura 10

Diagrama de bloques general del sistema



### 3.6.1.1. Sistema de entrenamiento

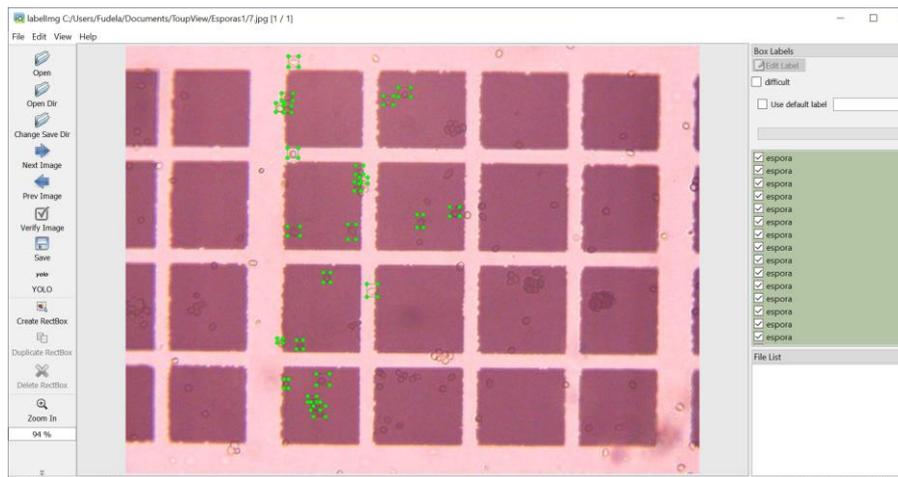
El sistema de entrenamiento se realiza en los servidores “Colab” de Google debido a que ofrecen una gran capacidad de procesamiento y un tiempo de uso libre, estas acciones se realizan una única vez y posteriormente sus resultados serán utilizados en el proceso de conteo de esporas en la placa embebida.

En la Figura 14 se muestra el diagrama de bloques para el entrenamiento del sistema que se divide en bloques como son: alimentación eléctrica, bloque de adquisición, bloque de preprocesamiento, base de datos de imágenes, bloque de convolución y bloque de entrenamiento. A continuación, se detalla el comportamiento de cada uno.

- Bloque de alimentación eléctrica: define las formas en las que se alimenta el sistema electrónico específicamente el hardware del sistema, siendo la cámara especializada para microscopio OMAX.
- Bloque de adquisición: sección donde se adquiere una imagen de esporas microscópicas de hongo trichoderma mediante una cámara Omax especializada para microscopios.
- Bloque de preprocesamiento: actividad realizada en un computador, en este caso una laptop Toshiba c-55, se basa en la detección de bordes de las esporas de hongo trichoderma y su etiquetación, adicional se realiza mediante código con Python la separación de la base de datos para obtener imágenes tanto para el entrenamiento como para realizar pruebas del sistema, respetando la división de 80% y 20% respectivamente de la muestra de 150 imágenes. A continuación, los procesos mencionados en este párrafo:
  - Detección de bordes: esta técnica se basa en delimitar todas las esporas encontradas en una imagen en un cuadro de pixeles para mejorar el resultado de la detección del objeto (espora trichoderma).
  - Etiquetado: esta acción se basa en brindar etiqueta de la clase que se desea identificar, en este se denomina “espora”, con un promedio de 88 esporas por imagen existen aproximadamente 28160 esporas etiquetadas para la base de datos. Para estos dos procedimientos se utiliza la aplicación de código abierto “LabelImg”, se observa en la Figura 11 junto con quince esporas encerradas en un cuadro y con sus etiquetas.

Figura 11

### Aplicación LabelImg

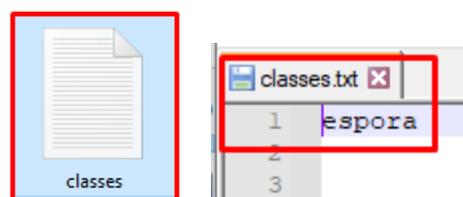


*Nota: Este programa permite delimitar cada uno de los objetos que se desea identificar, en este caso esporas de hongo trichoderma, y añadir una etiqueta a cada una de las delimitaciones. Para su correcto funcionamiento esta etiqueta debe ser la misma que se coloque en la red neuronal y el nombre del archivo resultante debe ser igual al nombre de la imagen en la que se basa.*

Al terminar de etiquetar por completo las esporas de una imagen este programa brinda dos archivos de texto donde el archivo denominado “classes.txt” especifica las clases que se han etiquetado (en este caso solo la clase “espora”), como se observa en la Figura 12. Cabe recalcar que este archivo solo se crea una vez.

Figura 12

*Archivo de etiquetado classes.txt*



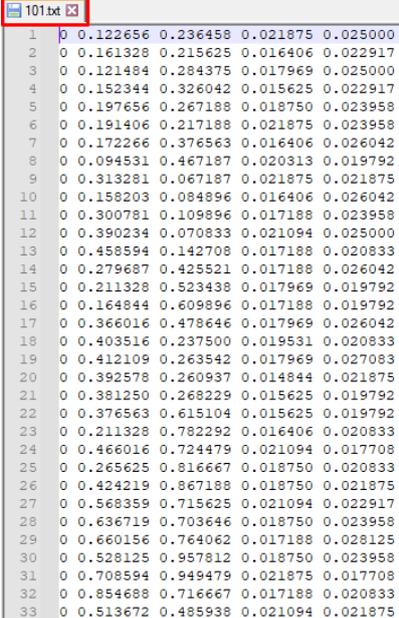
El segundo archivo que se crea tiene el mismo nombre que la imagen (en este caso las imágenes están numeradas del 1 al 152) y describe cada una de las etiquetas realizadas en la imagen en donde se observa la clase a la que pertenece y las coordenadas de los 4 puntos del cuadrado que encierra al objeto (en este caso una espora).

En la Figura 13 se observa el archivo de etiquetación perteneciente a la imagen “101”, por lo que este archivo lleva de nombre “101.txt”, la primera fila especifica la clase o etiqueta de cada objeto (espora) observamos que todas pertenecen a 0,

representando a la primera columna hallada en el archivo “classes.txt” siendo esta etiqueta “espora”.

Figura 13

Contenido de archivos etiquetación



Line	Col 1	Col 2	Col 3	Col 4	Col 5
1	0	0.122656	0.236458	0.021875	0.025000
2	0	0.161328	0.215625	0.016406	0.022917
3	0	0.121484	0.284375	0.017969	0.025000
4	0	0.152344	0.326042	0.015625	0.022917
5	0	0.197656	0.267188	0.018750	0.023958
6	0	0.191406	0.217188	0.021875	0.023958
7	0	0.172266	0.376563	0.016406	0.026042
8	0	0.094531	0.467187	0.020313	0.019792
9	0	0.313281	0.067187	0.021875	0.021875
10	0	0.158203	0.084896	0.016406	0.026042
11	0	0.300781	0.109896	0.017188	0.023958
12	0	0.390234	0.070833	0.021094	0.025000
13	0	0.458594	0.142708	0.017188	0.020833
14	0	0.279687	0.425521	0.017188	0.026042
15	0	0.211328	0.523438	0.017969	0.019792
16	0	0.164844	0.609896	0.017188	0.019792
17	0	0.366016	0.478646	0.017969	0.026042
18	0	0.403516	0.237500	0.019531	0.020833
19	0	0.412109	0.263542	0.017969	0.027083
20	0	0.392578	0.260937	0.014844	0.021875
21	0	0.381250	0.268229	0.015625	0.019792
22	0	0.376563	0.615104	0.015625	0.019792
23	0	0.211328	0.782292	0.016406	0.020833
24	0	0.466016	0.724479	0.021094	0.017708
25	0	0.265625	0.816667	0.018750	0.020833
26	0	0.424219	0.867188	0.018750	0.021875
27	0	0.568359	0.715625	0.021094	0.022917
28	0	0.636719	0.703646	0.018750	0.023958
29	0	0.660156	0.764062	0.017188	0.028125
30	0	0.528125	0.957812	0.018750	0.023958
31	0	0.708594	0.949479	0.021875	0.017708
32	0	0.854688	0.716667	0.017188	0.020833
33	0	0.513672	0.485938	0.021094	0.021875

- Bloque base de datos: etapa que tiene el objetivo de recolectar varias imágenes de esporas de hongo trichoderma y su respectiva delimitación de bordes para cada espora, hasta llegar a un total de 150 imágenes. Respetando la división de la base de datos.
- Bloque de convolución: realizado en el servicio de procesamiento en nube llamado “Google Cloud”, es el encargado de todos los procesos de convolución necesarios para el proceso de CNN. En la Figura 13, se observan diferentes parámetros como: filtros en donde se describe cuantas veces se repetirá la convolución y la red convolucional denominada “darknet” de la cual desciende el sistema de detección o reconocimiento de objetos “Yolo”, en el presente proyecto se usa una variable más ligera denominada “Yolo tiny”.
- Bloque de entrenamiento: se hace uso de la base de datos de entrenamiento, los mapas de caracteres dados por el bloque de convolución y la red convolucional denominada “darknet” es el bloque encargado de todos los procesos de entrenamiento del sistema y su resultado son los pesos del modelo pre entrenado.

La Figura 15 muestra el diagrama de flujo del proceso de entrenamiento del sistema de detección y conteo de esporas trichoderma, el cual especifica las acciones

necesarias para la creación del mapa de caracteres necesario para el aprendizaje por herencia que se usará en el siguiente bloque de detección y conteo.

Figura 14

Diagrama bloque entrenamiento

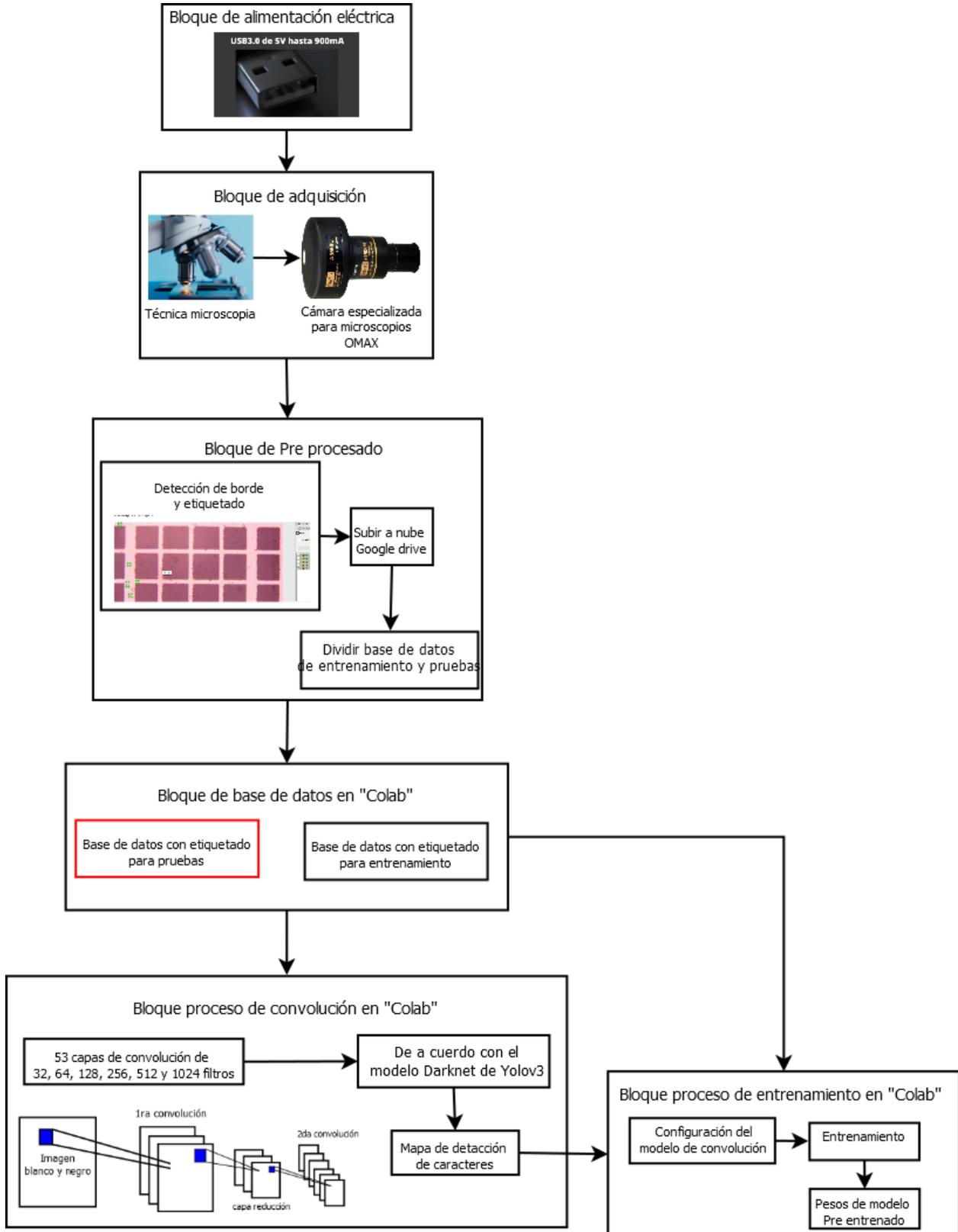
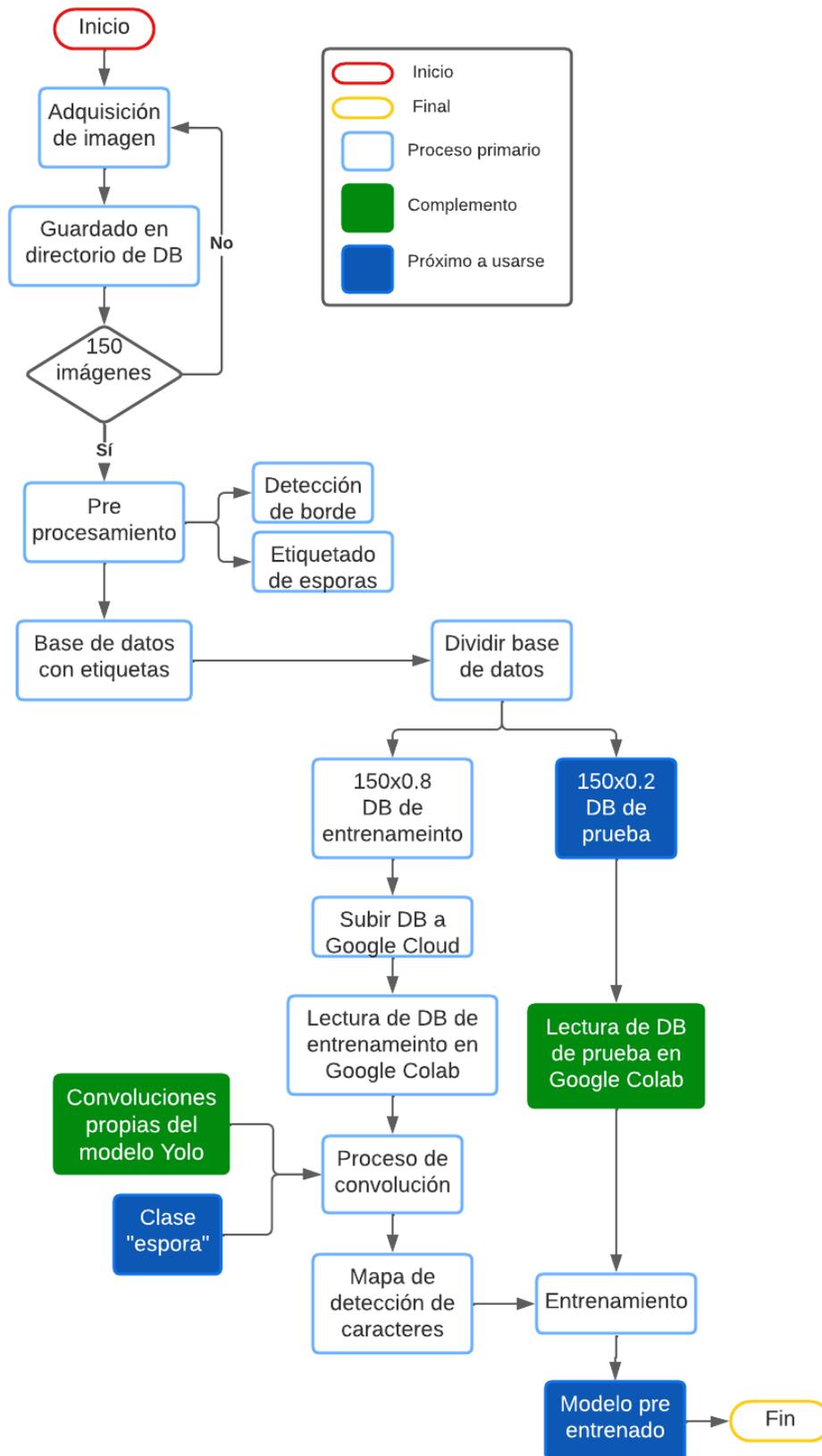


Figura 15

Diagrama flujo para el entrenamiento del sistema



### 3.6.1.2. Sistema de conteo de esporas trichoderma

En la Figura 16 se detalla el diagrama de bloque del funcionamiento del sistema, donde se divide en diferentes bloques los cuales son: alimentación eléctrica, bloque de adquisición, bloque de procesamiento CNN y bloque de visualización y almacenamiento. A continuación, se muestra las funciones de cada bloque:

- Bloque de alimentación eléctrica: define las formas en las que se alimenta el sistema electrónico específicamente el hardware del sistema, siendo la cámara especializada para microscopio OMAX y la placa embebida Nvidia Jetson Nano.
- Bloque de adquisición: sección donde se adquiere una imagen de esporas microscópicas de hongo trichoderma mediante una cámara Omax especializada para microscopios, siendo considerada como un paso fundamental tanto para brindar imágenes a la base de datos de aprendizaje como para la detección de las esporas de hongo trichoderma en esa fotografía. Se puede observar las siguientes etapas:
  - Primero se debe mencionar la técnica de visualización en microscopio electrónico que es la denominada luz transmitida de campo claro, ideal para organismos que permitan el paso de luz como las esporas del hongo trichoderma.
  - La cámara Omax viene con 3 acoples de los cuales uno de ellos permite la adaptación a la lente del microscopio (brindado por la empresa BiosebOrganics). Esta cámara tiene un sensor CMOS especializado para trabajar con diferentes técnicas de microscopía, mencionado en la Tabla 12, así mediante el proceso fotoeléctrico del sensor convierte la información analógica de una imagen en digital (en este caso de 5Mpx).
  - Mediante el programa ToupView y una conexión USB se obtiene la imagen de la cámara Omax, este dispositivo tiene la peculiaridad de cerrar el flujo de datos hacia otros programas por lo que no se puede tomarlas directamente de una interfaz hecha con el lenguaje de programación Python. Finalmente se guarda la imagen en un directorio dentro del sistema de la placa embebida Nvidia Jetson Nano.

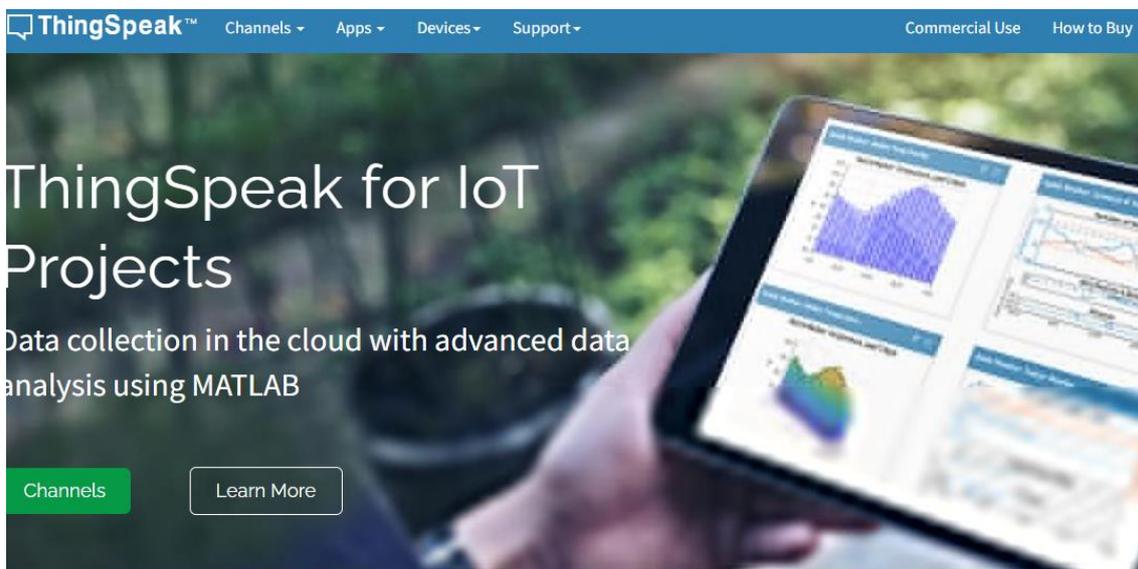
- Bloque de procesamiento: sección cuyo objetivo es realizar procesos de aprendizaje profundo CNN redes neuronales convolucionales haciendo uso del modelo pre entrenado obtenido del proceso de entrenamiento, el sistema de detección de objetos “Yolo tiny”, las clases solicitadas y mapa de detección de caracteres. Dando como resultado la detección de esporas en una imagen dada y su conteo. Se debe recalcar que se utiliza modelo pre entrenados creado en el software “Google Colab” para el entrenamiento y los procesos de convolución. Debido a que la placa embebida Nvidia Jetson Nano está construida para la ejecución de códigos de visión artificial incluso con aprendizaje profundo, pero no para el entrenamiento de la red. Para la red neuronal, ejecutada en la placa embebida, se utiliza el modelo de código libre denominado “Yolo tiny” con los componentes tomados del proceso de entrenamiento, como son el modelo pre entrenado y las clases a predecir (el etiquetado). A continuación, se profundiza en el modelo pre entrenado y la red neuronal.
  - Modelo Pre entrenado y etiquetación: El modelo pre entrenado, ejecutado en Google Colab, permite al sistema analizar el entrenamiento realizando procesos de convolución y comparaciones con la base de datos de entrenamiento con sus respectivas etiquetas y delimitaciones de borde de esporas del hongo trichoderma. La imagen de entrada es usada en el modelo pre entrenado leído por el código abierto mencionado y el modelo de detección de objetos “Yolo tiny”, el cual necesita información del modelo de características obtenido de las convoluciones especificadas en “Yolo tiny” y realizadas en Google Colab (usados para transferencia de aprendizaje). Este tiene el objetivo de identificar patrones entre los cuales están las formas básicas dados por procesos de convolución e identificación de esporas (clase) en la base de datos de entrenamiento, todo esto ubicado en el modelo de preentrenamiento.
  - Red neuronal: La red neuronal, ejecutada en la placa embebida, se encarga de la detección de esporas en la imagen adquirida utilizando la información leída del modelo pre entrenado por parte del modelo Darknet; usando tanto el mapa de características, los pesos del entrenamiento existentes del modelo pre entrenado y las clases especificadas (en este caso solamente la clase espora). Una vez realizada la identificación de la figura de la espora del hongo se la encierra en un cuadrado utilizando la librería Opencv, se

aumenta el valor de un contador en uno y se continua por toda la imagen hasta finalizar su revisión, esto mediante la librería YoloV3 que permite el uso de “Yolo tiny”.

- Bloque de visualización y almacenamiento: El objetivo es la visualización y almacenamiento de los resultados del sistema de identificación y conteo de esporas. Primero se visualiza el resultado en una interfaz gráfica creada en tkinter Python, para su almacenamiento utiliza una base de datos alojada en servidores públicos, para este fin se selecciona la plataforma IoT denominada Thing Speak. En la Figura 16 podemos observar la página de inicio de esta plataforma.

Figura 16

Página de inicio de plataforma IoT



*Nota: Esta plataforma puede manejar la transferencia de datos mediante el protocolo HTTPs y una versión beta del protocolo MQTT.*

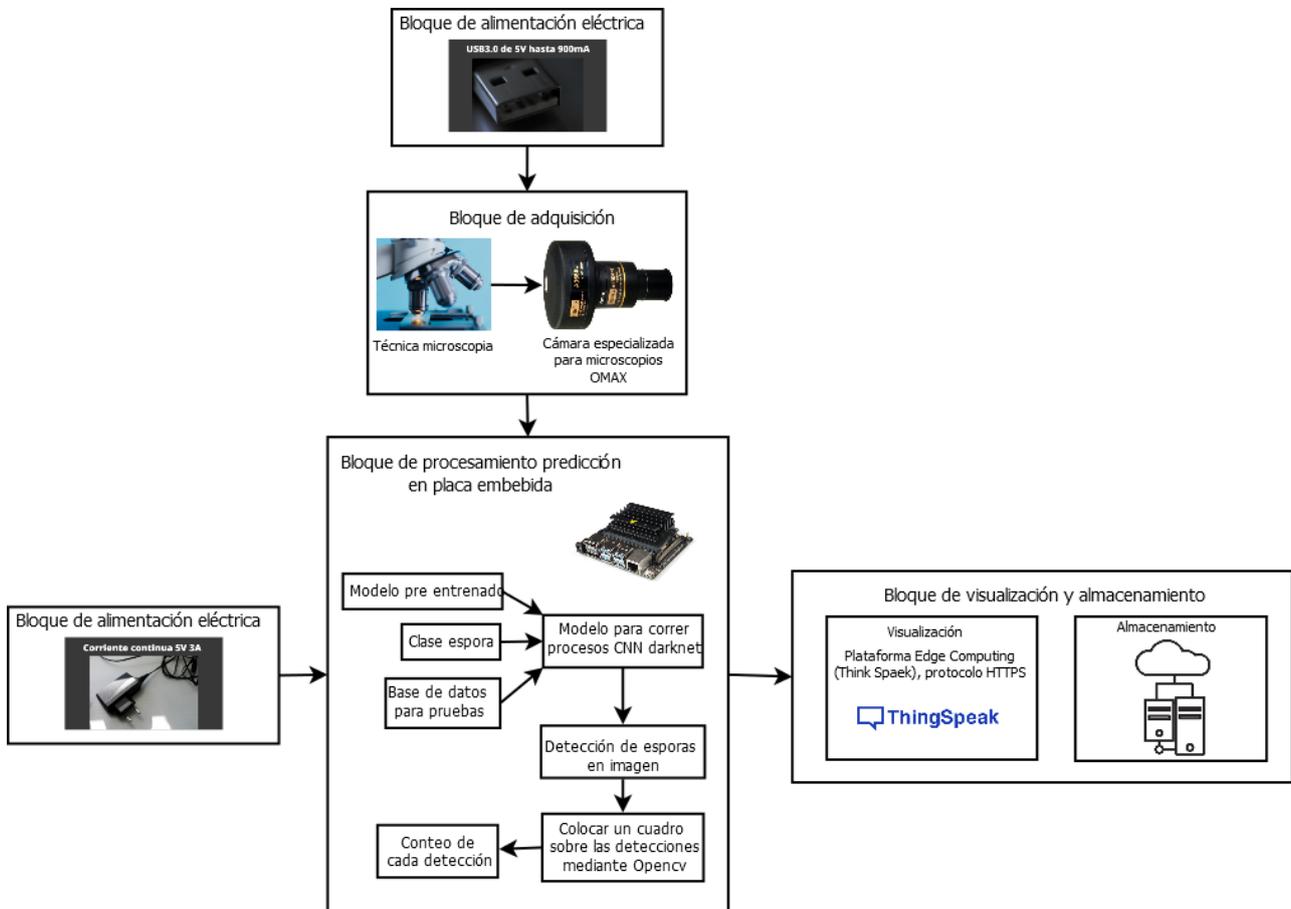
*Fuente: (The MathWorks, 2022)(IOTHOOK, n.d.)*

Se usa el protocolo para internet de las cosas HTTPs de uso libre. En el presente proyecto el cliente es la placa embebida Nvidia Jetson Nano y los servidores son parte de la aplicación en nube Thing Speak.

A continuación, en la Figura 17, se presenta el diagrama de bloques perteneciente al sistema de conteo de esporas de hongo trichoderma, mostrando las etapas explicadas en esta sección.

Figura 17

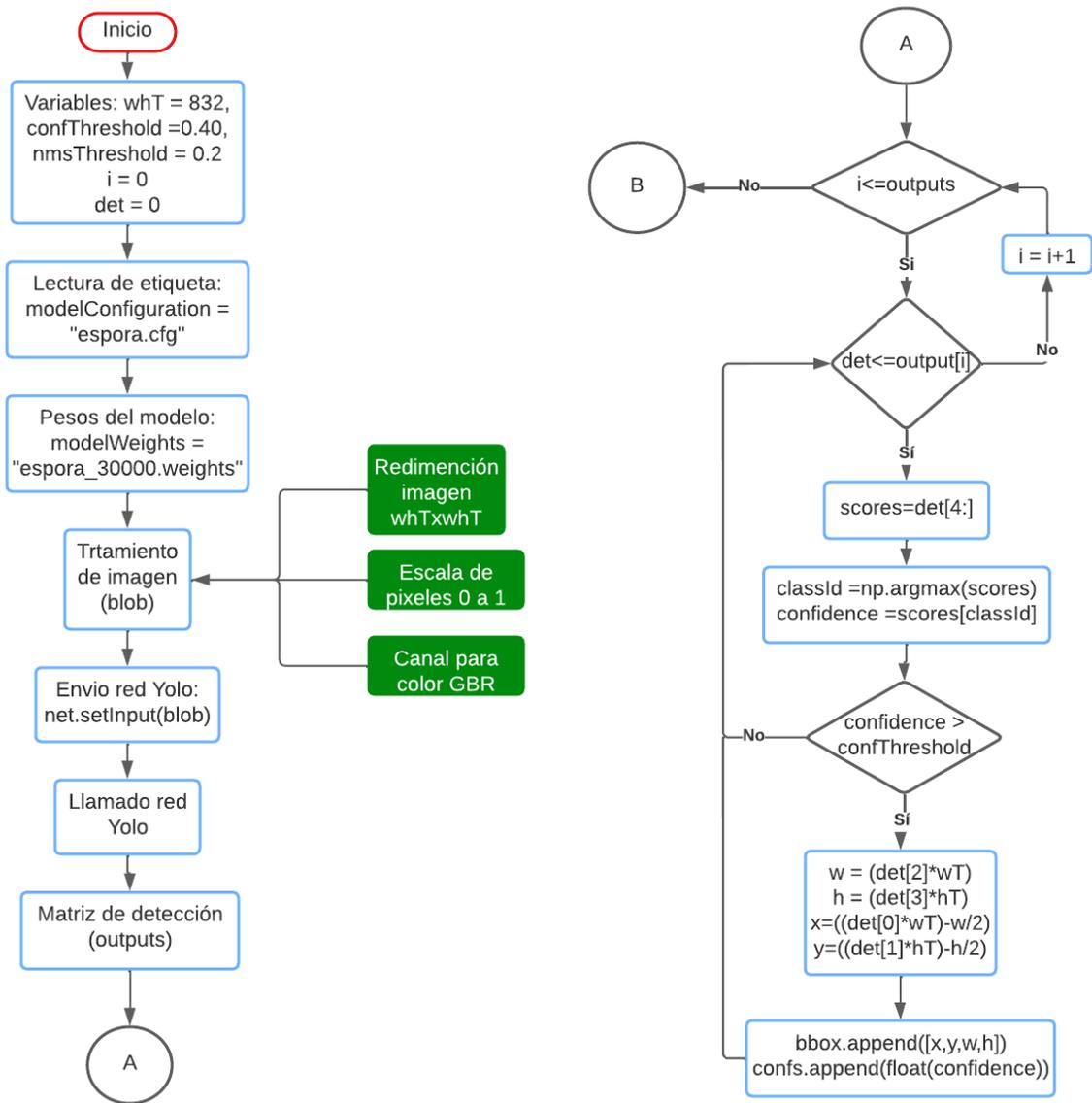
Diagrama de bloque de funcionamiento

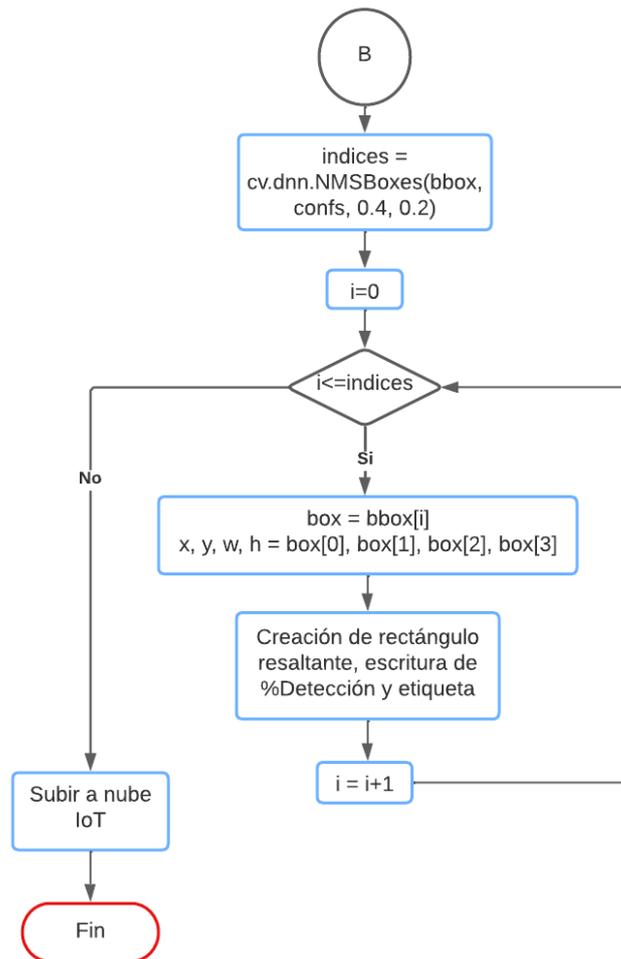


La Figura 18, se muestra el diagrama de flujo del sistema de conteo de esporas trichoderma el cual especifica los procesos que se accionan al iniciar el sistema, comenzado al iniciar los dispositivos, siendo estos: placa embebida “Nvidia Jetson Nano” y cámara Omax la cual captura una imagen al acoplarse a un microscopio y se almacenan en un directorio específico de la placa embebida, para posteriormente ser adquirido por una interfaz gráfica creada con el lenguaje de programación de alto nivel Python en donde, mediante la red neuronal convolucional “darknet” se lee el modelo pre entrenado, el etiquetado obtenido del sistema de entrenamiento y el mapa de caracteres para finalmente realizar el proceso de detección y conteo de esporas en la imagen con la ayuda de “Yolo tiny (You Only Look Once tiny)”.

Figura 18

Flujograma sistema de conteo y detección





Se presenta el algoritmo del sistema de detección y conteo del proyecto:

1. Obtener imagen
2. Brindar a red Yolo mapa de caracteres y configuración de capas de entrenamiento
3. Imagen a matriz de 3-dimenciones
  - a. Redimensión escala de imagen (1/255)
  - b. Modificación del tamaño (416x416)
  - c. RGB hacia BGR
  - d. Evitar el recorte de la imagen
4. Extraer operaciones de entrenamiento en variable "LayersNames"
5. Recorrer: `LN [i [0] -1] for i in LayersNames`, 2 tipos de capas de salida (yolo\_16 y yolo\_23) relacionadas con dos escalas (26x26 y 13x13).
6. Reenvío a red Yolo de LayersName (`forward(outputNames)`).

7. Obtener matriz “outputs”, [(2028,6) y (8112,6)]. Donde 2028 y 8112 representa todas las predicciones y 6 es el número de elementos en una entrada de la matriz: [x, y, w, h, %Confianza, categoría].
8. Recorrer mediante 2 ciclos la matriz “outputs” guardar de cada entrada a partir de la posición 4.
  - a. Obtener valores más altos del % Confianza.
  - b. If %Confianza > 0.4 (40%)
    - i. Obtener coordenadas (x,y,w,h) en variable bbox
9. Eliminar exceso de rectángulos resultantes con una objetividad menor que “0.2” (Tiny Yolo mide la objetividad mediante la operación intersección sobre unión de los cuadros) mediante el comando NMSBoxes, guardar en variable “index”
10. Recorrer “index”, graficar mediante OpenCv
11. Uso de len (index) como contador
12. Subir información de contador a nube

### 3.7. Costos asociados al desarrollo del prototipo

La Tabla 17 tiene una lista de componentes software requeridos para el sistema, donde se contemplan sus nombres, el estado de sus licencias de uso y la función en el sistema:

Tabla 17

*Componentes de software*

Software	Costo	Función
Visualizador de imágenes para la cámara OMAX ToupView	Uso libre	Capturar imágenes de las muestras de mezcla con hongos trichoderma mediante la cámara OMAX y un microscopio electrónico.
Python	Uso libre	Leguaje de programación de alto nivel, donde se desarrolla la técnica de visión artificial, recolección y tratamiento de imágenes.
Redes neuronales convolucionales (CNN)	Uso libre	Procesamiento de reconocimiento de esporas de hongo trichoderma mediante convoluciones de imagen y redes neuronales.
Jupyter Netbook IDE	Uso libre	Entorno compatible con Python para programación del código

Google Drive	Uso libre	Guardado de base de datos de entrenamiento con su delimitación de borde y etiqueta
Google Colab	Uso libre	Ejecución del bloque de entrenamiento del sistema, incluye procesos de convolución

La Tabla 18 muestra un listado de componentes de tipo hardware del presente proyecto, en donde se observa el número de componentes requeridos, el nombre de la pieza y su función.

Tabla 18

*Componentes físicos*

#	Componente	Función
1	Placa embebida Jetson Nano	Placa embebida encargada de ejecutar el proceso de detección y conteo de esporas de hongo trichoderma, además de la tarea de adquisición de imagen del microorganismo.
1	OMAX cámara USB microscopio	Su función principal es la captura de imagen del microorganismo a través del microscopio especializado.
1	Microscopio electrónico	Observación de microorganismo (incluido las esporas de hongo trichoderma).
1	Fuente de alimentación	Artefacto rectificador y transformador de energía eléctrica especializado para la placa Nvidia Jetson Nano.
1	Protección para placa embebida	Protección contra la manipulación de la placa embebida.
1	Capturadora de video	Permite transmitir video mediante el puerto HDMI de la placa embebida hacia el puerto USB de una laptop

---

150	Muestras de la mezcla con hongo trichoderma	Parte líquida del lugar de cultivo de hongos trichoderma, con una buena cantidad de sus esporas.
1	Laptop o pantalla con entrada HDMI	Proyección gráfica y local del sistema d detección y conteo de esporas trichoderma.
1	Conexión a internet mediante tarjeta USB Wi-Fi	Conexión a internet para comunicación con sistema IoT.
1	Cable HDMI	Cable para conexión de placa embebida hacia una salida de video (pantalla o laptop).

---

## **4. CAPÍTULO IV. Implementación y pruebas de funcionamiento**

El capítulo IV se realiza en base a la fase control de calidad mediante el modelo de gestión de proyectos PMBOK, en donde se registran los resultados de las pruebas tanto funcionales como de producción, para evaluar el desempeño del sistema (Project Management Institute, 2017). Así mismo se verifican los entregables del proyecto como son: las mediciones de control de calidad y requerimientos verificados en relación con las exigencias de los Stakeholders.

### **4.1. Código del sistema**

La siguiente sección se enfoca en la descripción del código usado para el funcionamiento del sistema embebido para el conteo de esporas de hongo trichoderma con técnicas de visión artificial. Esto se divide tanto en el bloque de entrenamiento realizado en el software de nube “Google Colab” como en el bloque de funcionamiento ejecutado en la placa embebida “Nvidia Jetson Nano”.

#### ***4.1.1. Código del sistema de entrenamiento***

Código ejecutado en el software “Google Colab” el cual brinda la opción de usar el lenguaje de programación de alto nivel Python y procesamiento libre por veinte y cuatro horas. Se comenzará observando las líneas necesarias para el montado de la base de datos de aprendizaje con sus respectivas etiquetas ubicadas en Google Drive con el entorno Google Colab, posteriormente se evidencia los procesos de convolución y entrenamiento del sistema para la creación del modelo pre entrenado.

El fin de este procedimiento es obtener un archivo especificando los pesos para la red neuronal y mapas de caracteres. Este algoritmo se realiza en el software Google Colab debido a que la placa embebida Nvidia Jetson Nano no está diseñada para este tipo de algoritmos. En la Figura 19 se muestra la capacidad que brinda el sistema “Google Colab”.

Figura 19

Tarjeta gráfica de Google Colab

```
!nvidia-smi
Fri Mar 18 13:18:38 2022
+-----+
| NVIDIA-SMI 460.32.03      Driver Version: 460.32.03      CUDA Version: 11.2      |
+-----+-----+-----+-----+-----+-----+
| GPU  Name                   Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf           Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
|                                           |              |                  |
+-----+-----+-----+-----+-----+-----+
|   0   Tesla K80                 Off          | 00000000:00:04:0 Off |                    |
| N/A   53C    P8              32W / 149W |  0MiB / 11441MiB |          0%      Default |
|                                           |              |                  |
+-----+-----+-----+-----+-----+-----+
|
| Processes:
|  GPU   GI    CI          PID    Type   Process name                  GPU Memory
|        ID    ID                                   |          Usage              |
+-----+-----+-----+-----+-----+-----+
| No running processes found
+-----+-----+-----+-----+-----+-----+

```

NOTA: La tarjeta gráfica se observa al ejecutar el comando: “!nvidia-smi”, este modelo es Tesla-k80 con aproximadamente 12Gb de memoria.

#### 4.1.1.1. Sincronizar base de datos

Como primer paso se debe importar la librería “drive” existente en Google Colab para el montaje de la carpeta con el comando: `drive.mount("/content/drive")`. A continuación, en la Figura 20 se observa las líneas de código y el resultado de su ejecución:

Figura 20

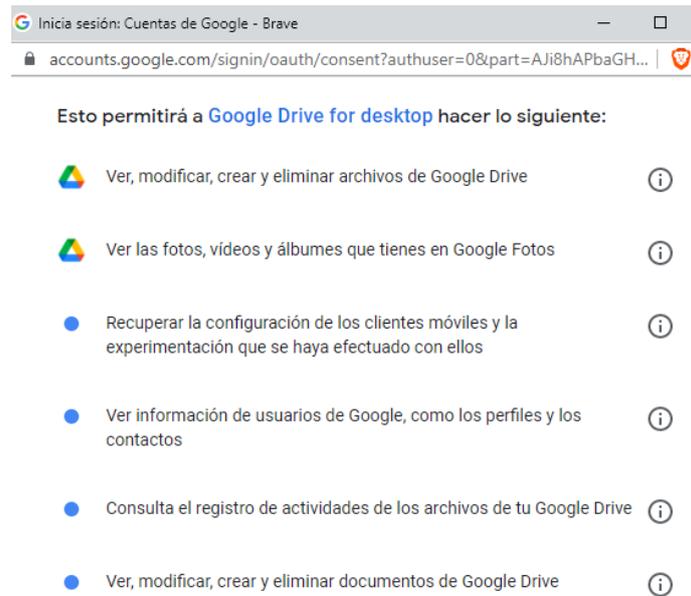
Montaje directorio de Google Drive

```
from google.colab import drive
drive.mount("/content/drive")
Drive already mounted at /content/drive
```

Al ejecutar el código perteneciente a la Figura 21 se pide brindar los permisos necesarios para el uso de Google Drive de los cuales el de mayor importancia es el permitir ver, modificar, crear archivos desde Google Drive.

Figura 21

Permisos para uso de Google Drive



Se debe cambiar el directorio de trabajo para encontrar aquel que contiene a la base de datos de aprendizaje con su respectiva etiqueta, esto se realiza con el módulo del sistema “os” que permite manipular la estructura de directorios. Se observa en la Figura 22 el funcionamiento de este cambio de directorio.

Figura 22

Cambio de directorio

```
import os #acceder directorios equipo
os.chdir("/content/drive/MyDrive/basedatos")
```

Guardar en variable las direcciones pertenecientes a un directorio (train/img) con las imágenes de la base de datos de entrenamiento y otro perteneciente a la ruta donde se ubican los archivos con la delimitación de borde y etiquetado. En la Figura 23 se observa el código utilizado para esta acción.

Figura 23

Guardar ruta de base de datos de entrenamiento en variables

```
# direcciones carpetas de objetos de entrenamiento
train_img_dir=os.path.join("train/img")
train_label_dir=os.path.join("train/label")
```

#### 4.1.1.2. Proceso de entrenamiento

Se comienza con la modificación del archivo de configuraciones del modelo “Darknet” para que este pueda usar la librería OpenCv, utilice la tarjeta GPU brindada por los servidores de Google Colab y permita usar la librería “CuDNN” perteneciente a Nvidia cuya función es acelerar el procesamiento en una tarjeta GPU como se muestra en la Figura 24.

Figura 24

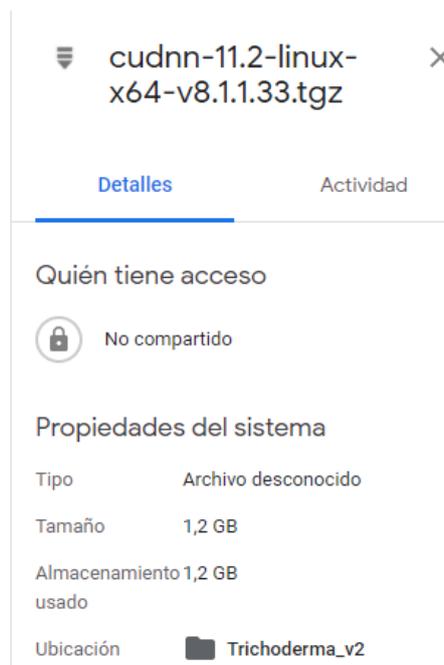
Modificación de archivo de configuración

```
!sed -i 's/OPENCV=0/OPENCV=1/' Makefile
!sed -i 's/GPU=0/GPU=1/' Makefile
!sed -i 's/CUDNN=0/CUDNN=1/' Makefile
!make
```

Se utiliza el software de código abierto de redes neuronales denominado “Darknet”, a continuación, se muestra el archivo descargado desde el repositorio de GitHub dentro del directorio Trichoderma\_v2 en la Figura 25.

Figura 25

Archivo comprimido de Darknet



Siguiendo con la exportación de esta carpeta mediante la librería “os” que nos da acceso al directorio en donde se encuentra la librería “Darknet” que en este caso es el directorio “Trichoderma\_v2”, como muestra la Figura 26.

Figura 26

Exportar directorio que contiene el código Darknet

```
import os
os.chdir("/content/drive/My Drive/Trichoderma_v2/")
```

Una vez dentro de este directorio se comprueba con el comando “ls” en la parte A de la Figura 27 y se procede a descomprimir en la parte B de la misma figura. Este archivo que contiene la programación relacionada al modelo “Darknet”, compatible con Yolo Tiny, se coloca los archivos resultantes en el directorio “/usr/local/”, hay que recalcar que este directorio pertenece a la nube de Google al igual que Colab.

Figura 27

Descomprimir archivo descargado

```
%ls
cuda/ include/ cudnn.h
cuda/ include/ cudnn_adv_infer.h
cuda/ include/ cudnn_adv_infer_v8.h
cuda/ include/ cudnn_adv_train.h
cuda/ include/ cudnn_adv_train_v8.h
cuda/ include/ cudnn_backend.h
cuda/ include/ cudnn_backend_v8.h
cuda/ include/ cudnn_cnn_infer.h
cuda/ include/ cudnn_cnn_infer_v8.h
cuda/ include/ cudnn_cnn_train.h
```

darknet\_v2/ train\_tricoderma.ipynb

```
|tar -xzvf /content/drive/My Drive/Trichoderma_v2/cudnn-11.2-linux-x64-v8.1.1.33.tgz -C /usr/local/
```

Continuando con el comando para ejecutar el entrenamiento mediante la arquitectura “Darknet” en donde se especifica la dirección de la base de datos de entrenamiento, base de datos de prueba y los nombres de las clases (en este caso la clase “espora”) todo esto se ubica en el archivo “espora.data”. En el archivo “espora.cfg” se encuentran los procesos de convolución y reducción propios de “Yolo tiny” siendo 13 convoluciones con diferentes tamaños de filtro y 6 reducciones denominadas maxpulling. También se proporciona un documento de entrenamiento con pesos previos (aplicando aprendizaje por herencia) propio del modelo “Darknet” denominado “darknet53.conv.74” el cual permite dar una base al entrenamiento.

Existe un segundo código debido a las limitaciones de tiempo que impone la versión gratuita de los servidores Google Colab, siendo muy similar al comando descrito, con la diferencia que se toma como base (aprendizaje por herencia) al último archivo de

entrenamiento que brindó este proceso denominado “espora.last.weights”. En la Figura 28 observamos estos comandos resaltando lo explicado.

Figura 28

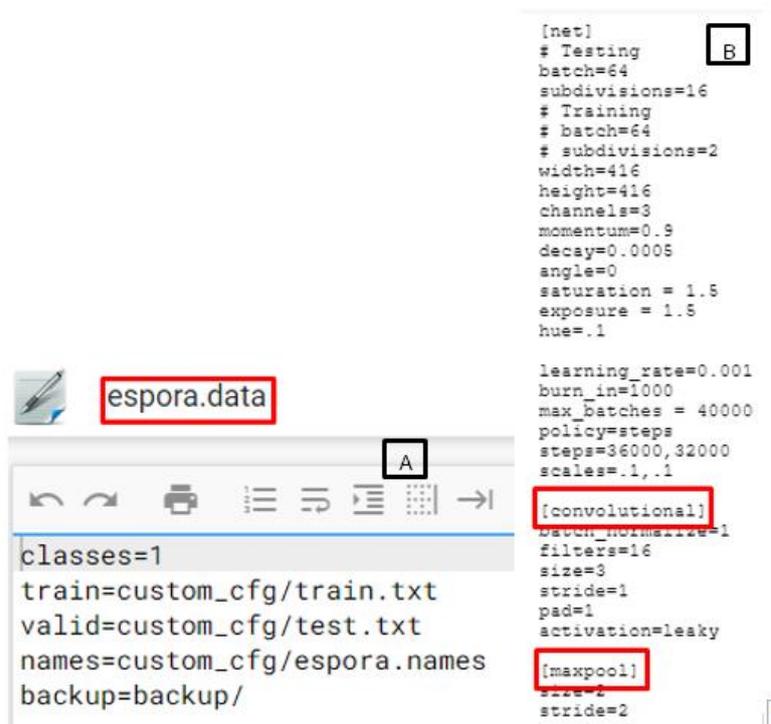
Comando para entrenamiento

```
#!/darknet detector train custom_cfg/espora.data custom_cfg/espora.cfg custom_cfg/darknet53.conv.74 -dont_show  
!./darknet detector train custom_cfg/espora.data custom_cfg/espora.cfg backup/espora_last.weights -dont_show
```

Como un adicional se muestra el contenido del archivo “espora.data” en la sección “A” y el contenido del archivo “esporas.cfg” en la sección B de la Figura 29. No se muestra el contenido del archivo de entrenamiento “darknet53.conv.74” debido a que este se encuentra encriptado.

Figura 29

Archivos espora.data y espora.cfg



#### 4.1.2. Código del sistema de conteo

A continuación, se describirá el código del sistema de detección y conteo de esporas de hongo trichoderma el cual es ejecutado a un nivel local en la placa embebida “Nvidia jetson nano”. Se empieza con el entorno de programación utilizado, en este caso el IDE “Spyder”, la librería para interfaces gráficas matriciales “tkinter”, librerías para

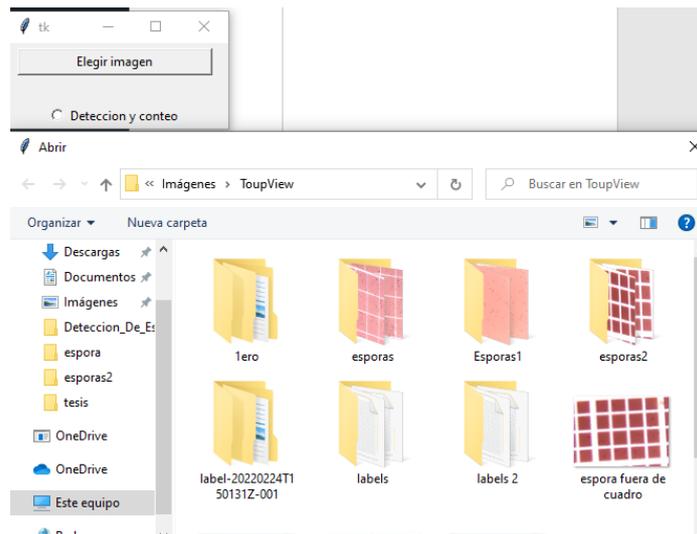
manipulación de imagen donde destaca “OpenCv”, procesos para preparación de imagen, aprendizaje por herencia (tomado del ítem 4.1.1) y pasos para iniciar la red neuronal convolucional “Yolo tiny”. A continuación, se presentan a más detalle dichos procesos:

#### 4.1.2.1. Adquisición de imagen

En la Figura 30 se coloca la plataforma gráfica para seleccionar imágenes, esta función es llamada al momento de accionar un botón que se explicará más adelante.

Figura 30

Selector de imagen



Aquí se muestra la función para la adquisición de imagen a través a de la interfaz gráfica (Figura 30) “elegir\_imagen”, el código se encuentra en la Figura 31. A continuación la descripción del proceso de este código:

- Se crea la variable “patch\_image” y los tipos de formatos admitidos los cuales son “jpeg, pnp y jpg” (formatos usados por la cámara Omax).
- A continuación, con una condicional “if” nos aseguramos de que la función se ejecute solamente cuando exista una imagen seleccionada.
- Se lee (mediante el comando `imread`) y se guarda en la variable global “image” la fotografía seleccionada, mediante la ayuda de la librería “OpenCV”.
- Se procede a mostrar la imagen, en la interfaz gráfica como fotografía de entrada, redimensionada a un ancho de 120Px mediante la librería “imutils”.
- Con el comando “`cv.cvtColor (imagen, cv.COLOR_BGR2RGB)`” se convierte los colores BGR a RGB permitiendo una manipulación de imagen con el formato RGB.

- Con el comando “Image.fromarray” se convierte de un array en formato numpy a formato PIL para ser graficado por Python y se guarda esta imagen en la variable “lblInputImage”.
- Posteriormente brinda la posición de la visualización.
- Para finalizar se colocan 2 líneas de código para eliminar la imagen procesada de salida al seleccionar otra fotografía diferente de esporas.

Figura 31

Código función de interfaz gráfica

```
def elegir_imagen():
    # Especificar los tipos de archivos, para elegir solo a las imágenes
    path_image = filedialog.askopenfilename(filetypes = [
        ("image", ".jpeg"),
        ("image", ".png"),
        ("image", ".jpg")])
    if len(path_image) > 0:
        global image
        global image1
        # Leer la imagen de entrada
        image = cv.imread(path_image)
        image1 = cv.imread(path_image)
        # Para visualizar la imagen de entrada en la GUI
        imageToShow= imutils.resize(image, width=120)
        imageToShow = cv.cvtColor(imageToShow, cv.COLOR_BGR2RGB)
        im = Image.fromarray(imageToShow)
        img1 = ImageTk.PhotoImage(image=im)
        lblInputImage.configure(image=img1)
        lblInputImage.image = img1
        # Label IMAGEN DE ENTRADA
        lblInfo1 = Label(root, text="IMAGEN DE ENTRADA:")
        lblInfo1.grid(column=0, row=1, padx=5, pady=5)
        # Al momento que leemos la imagen de entrada, vaciamos
        # la imagen de salida y se limpia la selección de los
        # radiobutton
        lblOutputImage.image = ""
        selected.set(0)
```

Nota: Este código es descrito al principio de la sección 4.1.2.1.

#### 4.1.2.2. Especificaciones de imagen y aprendizaje profundo

Se declara tres variables que se definen de la siguiente manera; whT” representa el largo y el ancho al cual se transforma la imagen, “confThreshold” es el umbral de precisión o coincidencia que se toma en cuenta al detectar una o varias esporas en cada matriz definida de pixeles y por último “nmsTheshold” representa el umbral de supresión no máxima cuya función es eliminar los cuadros candidatos redundantes y encontrar la mejor posición de detección de objetos suprimiendo elementos (cuadros) cuyo IOU sea mayor al umbral dado. Cabe recalcar que la cantidad IOU se refiere a Intersección sobre unión, es decir, la intersección de dos cuadros delimitadores divididos por su unión.

Se carga la etiqueta “espora” del archivo “*espora.names*” y se guarda en la variable “*classNames*”. A la variable “*modelConfiguration*” se le asigna la arquitectura de la red especificada en el archivo “*espora.cfg*” (donde se encuentra las convoluciones y reducciones realizadas en el entrenamiento en la nube Google Colab). En la variable “*modelWeights*” se cargan los pesos utilizando aprendizaje por herencia mediante el archivo “*espora\_30000.weights*” entrenado en los procesos observados en el ítem 3.6.1.2 (se repitió el entrenamiento un total de 30000 veces).

En la variable *net* se procede a cargar los pesos del modelo pre entrenado y las configuraciones de las convoluciones del mapa de características ya descargadas de los resultados obtenidos de “Google Colab”. En las variables “*modelConfiguration* y *modelWeights*” se brinda un complemento para usar la librería “OpenCV” además se especifica que el algoritmo de este programa se ejecute en la CPU de la tarjeta embebida. El código se presenta en la Figura 32.

Figura 32

Código de especificación de imagen y aprendizaje profundo

```
whT = 832 # ancho y alto de la imagen
confThreshold = 0.40 # umbral de precision o conincidencia
nmsThreshold= 0.2 # Umbral suprecion no maxima

#Cargamos las clases o etiquetas del detector
classesFile = "espora.names"
classNames = []
with open(classesFile, 'rt') as f:
    classNames = f.read().rstrip('\n').split('\n')

## Configuramos nuestra Red
modelConfiguration = "espora.cfg" #Archivo de arquitectura de La Red
modelWeights = "espora_30000.weights" # Pesos de La Red
net = cv.dnn.readNetFromDarknet(modelConfiguration, modelWeights) # Configuracion de La red
net.setPreferableBackend(cv.dnn.DNN_BACKEND_OPENCV) # configuramos opencv por detras
net.setPreferableTarget(cv.dnn.DNN_TARGET_CPU) # configuramos que trabaje con cpu
```

#### 4.1.2.3. Edición imagen y matriz de detección de esporas

Se carga la imagen llamando a la variable global “*image*” y calcándola en la variable local “*img*”, posteriormente se define un condicional para que la acción se ejecute solamente si se presiona el botón radial creado en el ítem 3.6.1.1 visto anteriormente.

Se realiza la preparación de la imagen (ubicada en la variable local *img*) mediante el código “*blob = cv.dnn.blobFromImage(img, 1/255, (whT, whT), [0, 0, 0], 1, crop=False)*”. Donde el primer parámetro (*img*) contiene la imagen adquirida, el segundo parámetro es factor de escala (*1/255*) que permite convertir cada píxel de la imagen a una

escala de 0 a 1, el tercero ( $whT$ ,  $whT$ ) es un redimensionamiento de la imagen. El cuarto parámetro ( $[0, 0, 0]$ ) es una matriz para poder procesar la imagen con sus capas de colores, el quinto ( $l$ ) es intercambio de RB o de color y el comando ( $crop=False$ ) permite que se reduzca la imagen y esta no se corte. Al final se brinda la variable ( $blob$ ) a la red neuronal de Tiny Yolo. Continuando con el código se tiene los siguientes procesos:

- La variable “*layersNames*” obtiene todas las predicciones de cada división de imagen realizado por Yolo Tiny, con la ayuda del comando “*net.getLayerNames*”.
- Se crea la variable “*outputName*” que contiene un array de todas las detecciones al recorrer “*layersNames*” con la ayuda del resultado del comando “*net.getUnconnectedOutLayers*”.
- Se envía “*outputName*” a la red neuronal de Tiny Yolo con el uso del comando “*net.forward*” y esto se guarda en la variable “*outputs*”.
- Para finalizar se brinda tanto la imagen tratada y las leyes de la red neuronal a la función “*findObjects*” explicada en el siguiente punto. En la Figura 33 se muestra el código de este proceso.

Figura 33

Código para función de edición de imagen y matriz de detección

```
def deteccion():
    global image
    if selected.get() == 1:
        img=image #Imagen
        blob = cv.dnn.blobFromImage(img, 1/255, (whT, whT), [0, 0, 0], 1, crop=False) #Tratamiento imagen
        net.setInput(blob) #enviar imagen tratada a red Tiny Yolo
        layersNames = net.getLayerNames() #Extraer operaciones de entrenamiento de red Yolo
        #print(net.getUnconnectedOutLayers())
        outputNames = [(layersNames[i[0]-1]) for i in net.getUnconnectedOutLayers()] #Recorrido de operaciones
        print(outputNames) #reenvio de operaciones de entrenamiento
        outputs = net.forward(outputNames) #obtencion matriz de deteccion [(2028,6) y (8112,6)].
        findObjects(outputs,img)
```

#### 4.1.2.4. Función para resaltar objetos encontrados

Se define la función denominada “*findObjects*” donde se alberga el código para resaltar objetos identificados y se definen las variables de entrada “*outputs*” (matriz de detección) e “*img*” (imagen adquirida). En la Figura 34 se coloca el código pertinente y se describe cada uno de sus componentes:

- El comando “*img.shape*” devuelve el número de filas, columnas y canales de “*img*” (brindado al llamado de esta función) las cuales son guardadas en las variables “*hT*, *wT* y *cT*”.

- Se realiza el recorrido de filas y columnas de la matriz “*output*” (brindado al llamado de esta función), guardando en la variable “*scores*” las cantidades del espacio 4 y 5 de la matriz resultante y en la variable “*det*” las cantidades de los espacios del 0 al 3 de la matriz (coordenadas de la detección).
- El comando “*classId = np.argmax(scores)*” devuelve el valor más alto del array “*scores*”, mediante la librería “Numpy”.
- En el próximo condicional observamos que si el umbral de coincidencia “*confidence*” (tomado de *classId*) es mayor a 0.4 (en nuestro caso) se procede con la siguiente acción:
  - Se brinda la medida para el cuadro que se dibuje tomando las coordenadas de la variable “*det*”, mediante la librería “OpenCV” y procede a dibujar estas coordenadas.
  - A continuación, en la variable “*indice*” con el comando “*cv.dnn.NMSBoxes*” y el umbral de supresión no máxima (*nmsThreshold*), se realiza la operación de supresión de cuadros resultantes con un umbral de supresión mayor a “*nmsThreshold (0.2)*”.
  - La variable “*esporas*” es el contador del sistema ya que mediante el comando “*len*” toma la cantidad de columnas en la matriz “*indices*” o el número de detecciones (en este caso mayores al 40%).
  - Para finalizar se grafica en un ciclo cada una de estas predicciones, colocando el porcentaje de detección y el nombre de la clase en cada resaltado creado (con la ayuda de la librería “OpenCV”).

Figura 34

Código para la función resaltar objetos

```
#funcion para dibujar los objetos encontrados y conexion con IOT
def findObjects(outputs,img):
    hT, wT, cT = img.shape #alto, ancho y canales de img
    bbox = []
    classIds = []
    confs = []
    for output in outputs: #recorrido 1
        for det in output: #recorrido 2
            scores = det[4:] #guardar en scores desde posicion 4 (%det,categoria)
            classId = np.argmax(scores) #sacar valores mas altos de scores
            confidence = scores[classId]
            if confidence > confThreshold: #(confidence>40%)
                w,h = int(det[2]*wT) , int(det[3]*hT) #medidas ancho y altura
                x,y = int((det[0]*wT)-w/2) , int((det[1]*hT)-h/2) #medidas coordenada x, y
                bbox.append([x,y,w,h])
                classIds.append(classId) #guardado %Det > 40%
                confs.append(float(confidence)) #guardado %Det > 40%

    indices = cv.dnn.NMSBoxes(bbox, confs, confThreshold, nmsThreshold)

    esporas=len(indices) #contador

    for i in indices:
        i = i[0]
        box = bbox[i] #recorrido de bbox
        x, y, w, h = box[0], box[1], box[2], box[3]
        cv.rectangle(img, (x, y), (x+w,y+h), (150, 150 , 150), 2) #creacion rectangulo
        cv.putText(img,f'{classNames[classIds[i]].upper()} {int(confs[i]*100)}%',
            (x, y), cv.FONT_HERSHEY_SIMPLEX, 0.45, (0, 255, 255), 1) #poner nombre de clase y %Det en

    cv.putText(img, 'Cantidad:'+str(esporas)+' esporas', (30,30), cv.FONT_HERSHEY_SIMPLEX, 1, (0,255,0),2)

    subir(str(esporas)) #operacion subir contador a nube
```

#### 4.1.2.5. Conexión hacia nube IoT

El código perteneciente a la función de subida de datos hacia plataforma IoT en la nube llamada Ting Speak toma la “llave de aplicación para escritura” que nos ofrece esta plataforma, se observa en la variable “data” en la Figura 35 (esta se oculta por razones de seguridad) y permite agregar la variable “esporas” que pide esta función (vista en el punto 4.1.2.4).

Figura 35

Código de función para subir información a nube

```
#####
def subir(esporas):
    #Conexion con sistema IoT
    data = urllib.request.urlopen("https://api.thingspeak.com/update?api_key= [REDACTED]+esporas);
```

## 4.2. Validación de requerimientos de Stakeholder

Se procede a realizar la validación de los requerimientos de los usuarios (principalmente técnicos de BiosebOrganics). Donde se observan los requisitos de importancia alta y media cumplidos en el proyecto, mientras que un requerimiento de bajo interés específicamente el SsRS4 (identificación de microorganismos invasores en la muestra de esporas trichoderma) no se cumple debido a la muy baja posibilidad de encontrar uno de estos. Cabe resaltar que en la etapa de recolección de muestras de esporas trichoderma en la base de datos del sistema no se encontró ningún microorganismo invasor. La Tabla 19 contiene este listado:

Tabla 19

Validación de requerimientos de Stakeholders

Características	Requerimientos de Stakeholders								
	SsRS1	SsRS2	SsRS3	SsRS4	SsRS5	SsRS6	SsRS7	SsRS8	SsRS9
Cámara Omax	SI								
Placa embebida Nvidia Jetson Nano con Case diseñado en 3D		SI							
Algoritmo de visión artificial en base a Tiny Yolo			SI						
Identificar microorganismos intrusos				NO					
Sistema de detección y conteo de esporas de hongo trichoderma					SI				
Adquisición de imagen mediante código en lenguaje Python						SI			
Técnica de visión artificial y aprendizaje profundo CNN (red neuronal convolucional)							SI		
Almacenamiento en nube mediante tecnología IoT								SI	
Pruebas de producción con duración de 2 semanas									SI

### 4.3. Pruebas funcionales

Se empieza con pruebas individuales referentes a la interfaz gráfica utilizada para el presente proyecto, se da inicio con la creación del gráfico del programa mediante la librería Tinker, esto se realiza con el fin de ofrecer una forma interactiva de utilizar el sistema. Se continúa analizando el código de la interfaz gráfica y haciendo pruebas individuales de su funcionamiento.

#### 4.3.1. Pruebas individuales interfaz gráfica

Se procede a realizar tres diferentes tipos de pruebas a la interfaz gráfica, las cuales son:

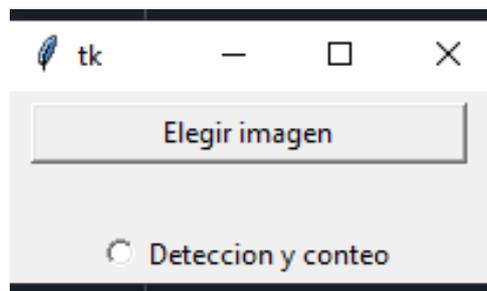
- Pruebas de funcionalidad de la interfaz donde se verifica si se adquiere correctamente la imagen mediante el uso de un botón de activación, si se activa el procedimiento de visión artificial al utilizar un botón radial y la visualización tanto de la imagen de entrada como la de salida (con sus respectivas detecciones).
- Pruebas de usabilidad del sistema en donde se indica la facilidad de uso de la interfaz.
- Pruebas de validación de la interfaz donde se comprueba los tres tipos de imagen que permite el código y observar que sucede si se intenta cargar un archivo de imagen con formato diferente.

##### 4.3.1.1. Pruebas funcionales de la interfaz gráfica

El primer avistamiento de la interfaz en donde se visualizan los botones de elección de imagen y el botón radial perteneciente a detección de esporas, como se observa en la Figura 36.

Figura 36

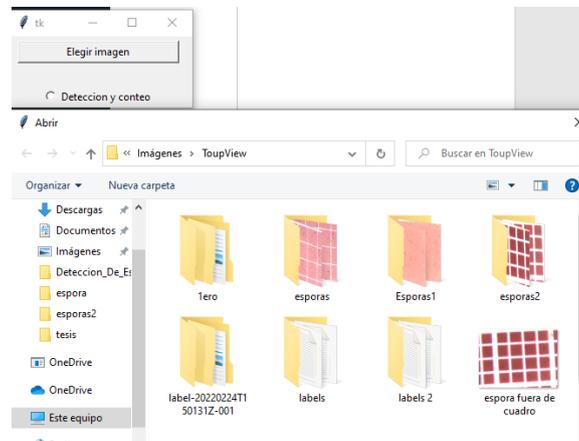
Primera vista aplicación gráfica del sistema de detección y conteo de esporas trichoderma



La primera prueba de funcionalidad es el botón de elegir imagen que nos despliega un navegador de directorios en donde se elige la fotografía de las esporas, código definido en la función de “elegir imagen” del punto 4.1.2.1. previamente explicado. La Figura 37 muestra el resultado de esta prueba.

Figura 37

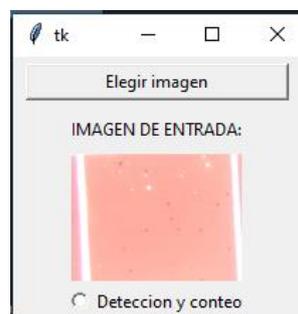
Selector de imagen



Una vez se selecciona la fotografía de la muestra de esporas de hongo trichoderma, esta imagen se muestra de forma reducida y con el mensaje de “Imagen de entrada”. Este segundo ítem de las pruebas de funcionalidad de la interfaz se observa en la Figura 38.

Figura 38

Imagen seleccionada de entrada



Para finalizar las pruebas de funcionalidad de la interfaz el usuario debe dar clic izquierdo en el botón de detección y conteo, esto dará inicio al proceso de conteo de esporas, al finalizar este proceso se graficará una nueva imagen que resalte las esporas encontradas y escribirá el total de esporas localizadas (dato que igualmente se sube hacia la plataforma de Edge Computing Thing Speak). En la Figura 39 se observa el resultado final y en la Figura 48 se observa cómo se muestran los datos en la plataforma en línea IoT tanto en forma de tabla y gráficas de datos.

Figura 39

Detección de esporas en interfaz gráfica



#### 4.3.1.2. Pruebas usabilidad de interfaz gráfica

Las pruebas de usabilidad se dividen en tareas para completar todo el proceso del sistema, posteriormente se brinda una comparación mediante una tabla de calificación. En la Tabla 20 se observa las ponderaciones de estas tareas, dando una calificación en base a las pruebas funcionales vistas en el ítem 4.3.1 A continuación, se observa el listado de tareas.

- Tarea 1: adquisición de la imagen
- Tarea 2: visualización de la imagen de entrada
- Tarea 3: inicio del proceso de visión artificial
- Tarea 4: Visualización de la imagen de salida con las respectivas detecciones y el conteo automático

Tabla 20

Tareas para usabilidad del sistema

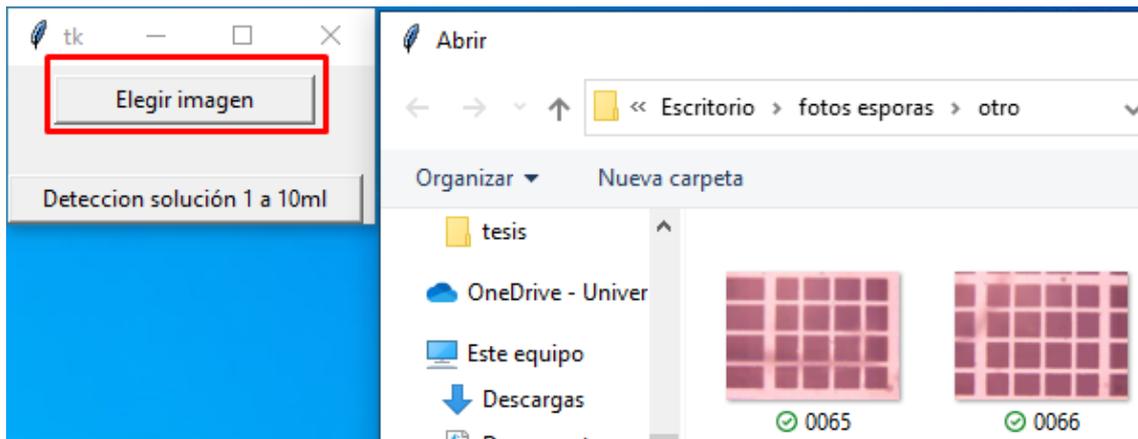
	¿Su uso es frecuente?	¿Es una tarea crítica?	¿Es importante para el sistema?	Total
	0 nada frecuente 5 muy frecuente	0 nada frecuente 5 muy frecuente	0 nada frecuente 5 muy frecuente	
<b>Tarea1</b>	5	5	5	15
<b>Tarea2</b>	1	1	1	3
<b>Tarea3</b>	5	5	5	15
<b>Tarea4</b>	5	5	5	15

### 4.3.1.3. Pruebas validación de interfaz gráfica

Las pruebas de validación se realizan al adquirir los formatos de imagen que permite el sistema los cuales son: jpeg, pnp y jpg. Además, se prueba el tomar diferentes archivos de imagen no concordantes como son tiff y bmp. En la Figura 40 en su sección B se mira el proceso gráfico para localización de imagen, mientras que en la sección A se observa el resultado de importar una imagen (en la interfaz gráfica del sistema).

Figura 40

Interfaz gráfica de adquisición de imagen



Se continua con la prueba de cada uno de los formatos compatibles (jpeg, pnp y jpg) y formatos no compatibles (bmp y tiff), observando los resultados en la Tabla 21.

Tabla 21

Pruebas de validación

Formatos	Funcionamiento
Jpeg	Correcto
Pnp	Correcto
Jpg	Correcto
.Bmp	No permite adquirir el formato Bmp
.Tiff	No permite adquirir el formato Tiff

### 4.2.1. Pruebas de aplicativo IoT

En esta sección se procede a comprobar si existe el envío de datos del aplicativo hacia la plataforma IoT en la nube. Posteriormente se ensaya el almacenamiento de datos

dentro de la plataforma IoT y por último se experimenta la visualización de estos datos (conteo total de esporas).

#### 4.2.1.1. Pruebas de envío de datos

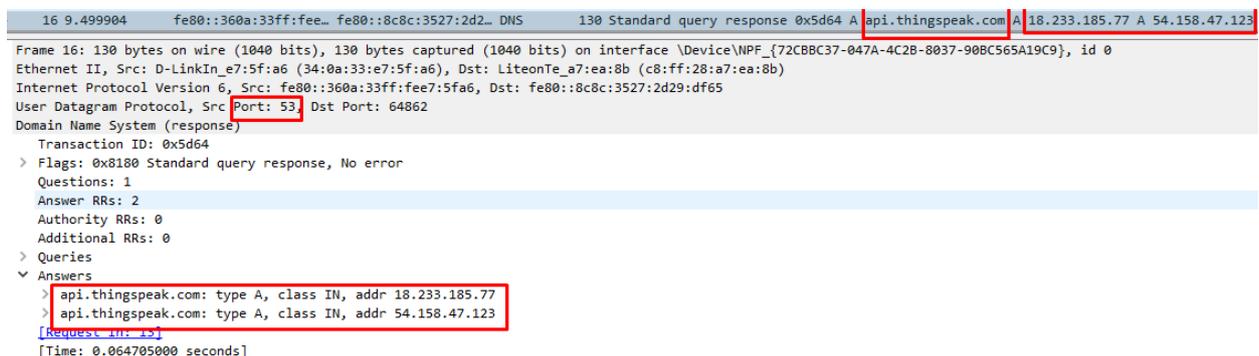
En esta sección se procede a verificar si existe un envío de datos a la plataforma IoT llamada Thing speak, mediante captura de tramas en el programa “WireShark”. En esta etapa se observa el protocolo HTTPs siendo utilizado para la transmisión de datos del conteo de esporas, viendo la conexión entre el sistema y la plataforma IoT desde la sincronización de las plataformas, conexión con el servidor DNS, certificados de seguridad TLS y cierre de sesión.

- **Conexión a la plataforma IoT**

La trama que se muestra en la Figura 41 representa la respuesta del servidor DNS, en este caso con la IP del dispositivo de salida a internet del actual proveedor ISP. Se deduce que se utiliza el puerto 53 con el protocolo TCP debido a que se utiliza el servicio DNS y el servicio IoT usa TCP. Se consulta el dominio “thingspeak.com” y se brinda las IPs públicas de la plataforma siendo estas: “18.233.185.77” o “54.158.47.123”, haciendo así posible la comunicación mediante la capa 3 de la arquitectura de red TCP/IP.

Figura 41

Trama respuesta DNS



```
16 9.499904 fe80::360a:33ff:fee... fe80::8c8c:3527:2d2... DNS 130 Standard query response 0x5d64 A api.thingspeak.com A 18.233.185.77 A 54.158.47.123
Frame 16: 130 bytes on wire (1040 bits), 130 bytes captured (1040 bits) on interface \Device\NPF_{72CB8C37-047A-4C2B-8037-90BC565A19C9}, id 0
Ethernet II, Src: D-LinkIn_e7:5f:a6 (34:0a:33:e7:5f:a6), Dst: LiteonTe_a7:ea:8b (c8:ff:28:a7:ea:8b)
Internet Protocol Version 6, Src: fe80::360a:33ff:fee7:5fa6, Dst: fe80::8c8c:3527:2d29:df65
User Datagram Protocol, Src Port: 53, Dst Port: 64862
Domain Name System (response)
Transaction ID: 0x5d64
> Flags: 0x8180 Standard query response, No error
Questions: 1
Answer RRs: 2
Authority RRs: 0
Additional RRs: 0
> Queries
< Answers
  > api.thingspeak.com: type A, class IN, addr 18.233.185.77
  > api.thingspeak.com: type A, class IN, addr 54.158.47.123
[Request: 111.13]
[Time: 0.064705000 seconds]
```

- **Establecimiento de conexión entre el sistema y la plataforma IoT**

En la Figura 42 se aprecia las tramas de sincronización de acuerdo con la teoría, donde existe una petición de sincronía por parte del cliente (sistema de detección y conteo automático) al servidor IoT mediante la trama “SYN”, se obtiene una confirmación del establecimiento de la comunicación con el paquete “SYN-ACK” para finalizar este proceso el cliente confirma recibir este SYN-ACK mediante un “ACK”.

Figura 42

Establecimiento de conexión entre el sistema y la plataforma IoT

174.50.100.119	18.233.185.77	TCP	66	54418 → 443 [SYN]	Seq=0 Win=64240 Len=0 MSS=1460 WS=256
18.233.185.77	174.50.100.119	TCP	66	443 → 54418 [SYN, ACK]	Seq=0 Ack=1 Win=26883 Len=0 MSS=1
174.50.100.119	18.233.185.77	TCP	54	54418 → 443 [ACK]	Seq=1 Ack=1 Win=131328 Len=0

- **Establecimiento de conexión seguro entre el cliente y el servidor**

El establecimiento de conexión seguro entre el cliente y el servidor muestra los paquetes pertenecientes al intercambio de certificado de seguridad privado. Para comprobar esto se observa la respuesta del servidor (cabe recalcar que se hace uso del puerto 443 perteneciente al protocolo de transferencia de hipertexto segura, también usado por la tecnología IoT). En la Figura 43 se observa los protocolos usados en este certificado (rsa, y sha512) y la respuesta del servidor IoT del intercambio de certificados de seguridad.

Figura 43

Respuesta de protocolo TSL del servidor

26	9.758068	18.233.185.77	174.50.100.119	TLSv1.2	194	Server Key Exchange	Server Hello Done
----	----------	---------------	----------------	---------	-----	---------------------	-------------------

[Frame: 26, payload: 207-337 (131 bytes)]  
[Segment count: 2]  
[Reassembled TCP length: 338]  
[Reassembled TCP Data: 160303014d0c00014903001741049f26f72b5d85c39ca8d775747b97bcb8e9ffd33dea51...]

Transport Layer Security

- ▼ TLSv1.2 Record Layer: Handshake Protocol: Server Key Exchange  
Content Type: Handshake (22)  
Version: TLS 1.2 (0x0303)  
Length: 333
  - ▼ Handshake Protocol: Server Key Exchange  
Handshake Type: Server Key Exchange (12)  
Length: 329
    - ▼ EC Diffie-Hellman Server Params  
Curve Type: named\_curve (0x03)  
Named Curve: secp256r1 (0x0017)  
Pubkey Length: 65  
Pubkey: 049f26f72b5d85c39ca8d775747b97bcb8e9ffd33dea51e4af63b0cb05e14cdfb120116e...
      - ▼ Signature Algorithm: rsa\_pkcs1\_sha512 (0x0601)  
Signature Hash Algorithm Hash: SHA512 (6)  
Signature Hash Algorithm Signature: RSA (1)  
Signature Length: 256  
Signature: 320dcf27c57f1680bee4e9b54215c0785824820662044aa3cc1e04b27966a3646af352b7...

Transport Layer Security

- ▼ TLSv1.2 Record Layer: Handshake Protocol: Server Hello Done  
Content Type: Handshake (22)  
Version: TLS 1.2 (0x0303)  
Length: 4
  - ▼ Handshake Protocol: Server Hello Done

#### 4.2.1.2. Pruebas de almacenamiento de datos

En la Figura 44 se muestran el guardado de 21 datos del conteo total de esporas del hongo trichoderma, en el formato de valores separados por coma brindado por la

plataforma IoT Thing speak. Archivo donde se encuentra la fecha exacta que estos se subieron, la hora de acuerdo con el horario ecuatoriano y para finalizar el dato del contador de esporas.

Figura 44

Guardado de datos

	A	B
1	created_at,entry_id,field1,latitude,longitude,elevation,status	
2	2022-06-22T09:10:26-05:00,1,068,,,,	
3	2022-06-22T09:16:49-05:00,2,036,,,,	
4	2022-06-22T13:14:11-05:00,3,013,,,,	
5	2022-06-22T13:15:12-05:00,4,020,,,,	
6	2022-06-22T13:16:10-05:00,5,025,,,,	
7	2022-06-22T13:16:48-05:00,6,036,,,,	
8	2022-06-22T13:18:15-05:00,7,036,,,,	
9	2022-06-22T13:18:51-05:00,8,021,,,,	
10	2022-06-22T13:19:21-05:00,9,032,,,,	
11	2022-06-22T13:59:30-05:00,10,012,,,,	
12	2022-06-22T14:00:19-05:00,11,013,,,,	
13	2022-06-22T14:00:53-05:00,12,012,,,,	
14	2022-06-22T14:01:21-05:00,13,023,,,,	
15	2022-06-22T14:06:17-05:00,14,031,,,,	
16	2022-06-22T14:22:58-05:00,15,012,,,,	
17	2022-06-22T14:26:16-05:00,16,027,,,,	
18	2022-06-22T14:26:39-05:00,17,015,,,,	
19	2022-06-22T14:27:13-05:00,18,020,,,,	
20	2022-06-22T14:27:40-05:00,19,020,,,,	
21	2022-06-22T14:28:04-05:00,20,012....	

#### 4.2.1.3. Pruebas de visualización

En la Figura 45 se muestra la visualización de datos en una gráfica estadística de la plataforma IoT considerando los parámetros del contador de esporas en relación con la fecha y hora de carga de datos. Al hacer contacto con uno de los puntos rojos que se miran en la gráfica se observa, de una forma amigable, el valor del conteo subido junto con su respectiva fecha y hora. Como ejemplo en la Figura 45 se encuentra esta gráfica junto con el conteo de 68 esporas trichoderma subidas el 22 de junio del 2022 a las 9:10 am.

Figura 45

Representación con tecnología Edge Computing



Nota:

La figura representa los datos guardados en tablas o base de datos

#### 4.2.2. Case protector

El diseño del armazón se hace en base a las medidas de la placa embebida y sus respectivos puertos, esto se mostró en el ítem 3.5.1.3 El case fue creado con material PLA (ácido poli-láctico) que es un polímero biodegradable derivado del ácido láctico siendo considerado un material altamente versátil echo de recursos renovables. El PLA presenta buenas propiedades físicas como un alto nivel de resistencia a la tracción y la ductilidad, esto hace que sea adecuado para diferentes medios de procesamiento, como el moldeo por inyección utilizado en varios tipos de impresora 3D (Rampérez Barraón, 2021). En la Tabla 22 se muestra sus propiedades mecánicas con la unidad de presión Mega Pascal.

Tabla 22

Propiedades mecánicas generales PLA

<b>Módulo elástico</b>	3000 – 4000 MPa
<b>Resistencia a la tracción</b>	50 – 70 MPa
<b>Resistencia a la flexión</b>	4000 – 5000 MPa

Nota: tomado como referencias las propiedades del material al trabajo de (Rampérez Barraón, 2021)

En la Figura 46 se observa el armazón junto con la placa embebida “Nvidia Jetson Nano”.

Figura 46

Resultado final del armazón



### 4.3. Pruebas funcionales locales

Esta prueba consistió en verificar el resultado del conteo de espora trichoderma en las imágenes previamente capturadas, mismas que fueron usadas para comprobar el sistema antes de llevarlo al ambiente de producción

La evaluación funcional local consta de un total de 20 muestras fotográficas, tomada de las imágenes de la base de datos de pruebas. En la Tabla 23 se observa el conteo manual, el conteo automático, además de una comparación del tiempo que toma realizar cada conteo, (haciendo visible los beneficios del sistema del presente proyecto).

Tabla 23

Comparación de conteo automático del sistema y conteo manual a concentración (1/10) ml

#	Cuenta manual Número de Esporas	Tiempo conteo manual	Sistema de Conteo	Tiempo conteo automático	FP	VN
1	37	3.5min	36	3seg	-1	2
2	28	3.5min	25	3seg	-4	1
3	23	3.5min	21	3seg	-2	1
4	34	3.5min	32	3seg	-3	1
5	29	3.5min	26	3seg	-3	0
6	30	3.5min	28	3seg	-2	1
7	33	3.5min	32	3seg	-1	2
8	28	3.5min	27	3seg	-1	2
9	23	3.5min	20	3seg	-4	1
10	38	4min	37	3seg	-1	0
11	32	4min	31	3seg	-1	2

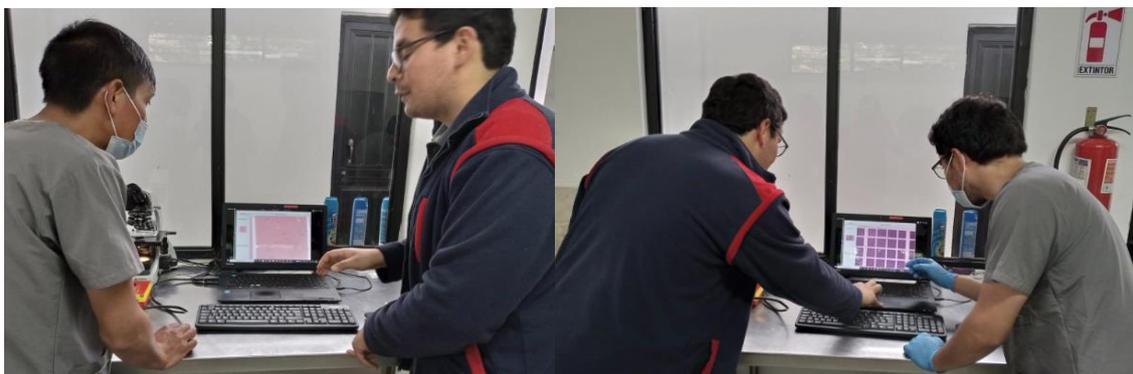
12	24	3.5min	23	3seg	-1	1
13	26	4min	26	3seg	0	0
14	71	4.3min	73	3.3seg	-3	5
15	80	4.4min	69	3.3seg	-12	1
16	81	4.4min	86	3.3seg	-2	7
17	67	4min	67	3.3seg	-3	3
18	88	4.4min	82	3.3seg	-7	1
19	81	4.4min	81	3.3seg	-3	3
20	89	4.4min	88	3.4seg	-2	1

#### 4.4. Pruebas de producción

En la empresa BiosebOrganisc se realizó una inducción al personal para el manejo correcto del sistema de detección y conteo automático, con fecha 15 de junio del 2022, donde se incluyó su manipulación y funcionamiento. Para constancia de esta capacitación se observa en el ANEXO F la lista firmada de asistencia de los cuatro beneficiarios (el jefe de plata Ing. Gabriel Curipallo y los laboratoristas: Ing. Mercy Escobar, Ing. Alejandro Ortiz e Ing. Manuel Bravo). Adicionalmente en la Figura 47 se colocan imágenes de este evento en donde se pueden observar a los ingenieros Manuel Bravo y Alejandro Ortiz respectivamente.

*Figura 47*

*Fotografía de la inducción al personal*



Las pruebas de producción se basan en la recopilación de datos tanto de conteo manual como el conteo automático del sistema en el laboratorio de Bioseb Organics y por el personal encargado y capacitado. Estas pruebas se realizaron en diferentes fechas, las cuales son:

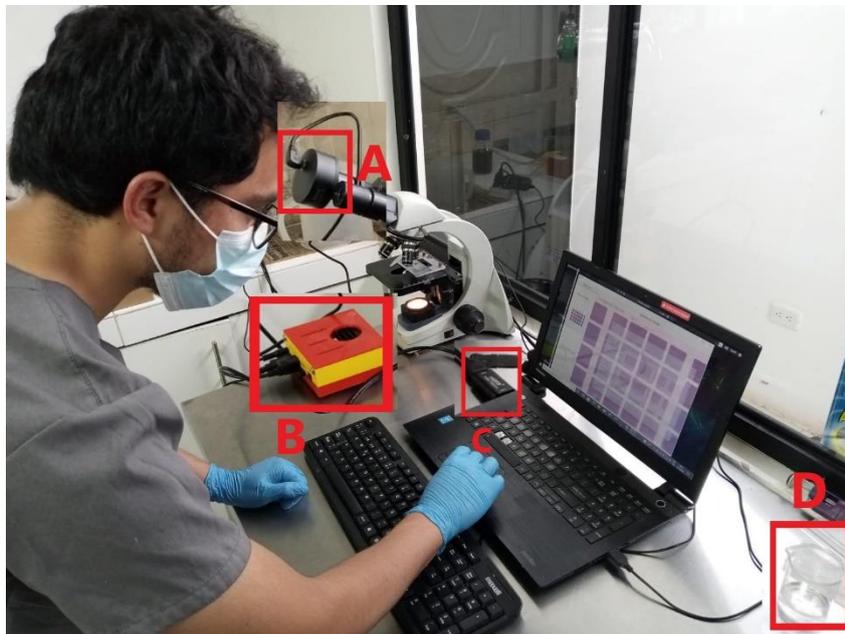
- **Semana 1:** miércoles 15 de junio a viernes 17 de junio del año 2022.
- **Semana 2:** lunes 20 de junio a viernes 24 de junio del año 2022.

- **Semana 3:** lunes 27 de junio a jueves 1 de julio del año 2022.

La Figura 48 representa el funcionamiento del sistema de detección y conteo de esporas de hongo trichoderma en su ambiente de producción (laboratorio de BiosebOrganics), en este caso a cargo del Ing. Alejandro Ortiz. En la imagen se observa la cámara Omax acoplada al microscopio en el cuadro A; la conexión de la placa Nvidia Jetson Nano (ítem B) a su fuente de energía, mouse inalámbrico, cámara Omax, entrada para pantalla tipo HDMI y entrada ethernet; la sección C es un dispositivo capturador de video que permite emitir la señal HDMI de la placa embebida a la pantalla de una laptop y el ítem D es una muestra de líquido con esporas de hongo trichoderma a una dilución a 1ml/10ml. Además, se aprecia gráficamente el sistema en la pantalla de la laptop.

*Figura 48*

*Fotografía del sistema en su ambiente de producción*



*Nota: A) cámara Omax. B) Placa embebida Nvidia Jetson Nano. C) Capturador de video HDMI a USB. D) Muestra de líquido de esporas de hongo trichoderma a una disolución de 1ml/10ml.*

Las muestras de esporas de hongo trichoderma se dividen en dos tipos de disoluciones: 1ml de sustancia directamente extraída de los hongos de género trichoderma en 10ml de líquido estéril y 1ml de sustancia directamente extraída de los hongos de género trichoderma en 100ml de líquido estéril. La principal diferencia entre estos conjuntos es la cantidad de esporas que aparecerán en la imagen del microscopio siendo obviamente mayor en la concentración 1ml/10ml y menor en la concentración 1ml/100ml.

Se debe aclarar que al tiempo de conteo tanto manual como automático hecho por el sistema, se debe aumentar el lapso que conllevan los procesos de enfoque y localización de las secciones correctas de la placa con el líquido que contiene las esporas de hongo trichoderma, estas acciones para el personal capacitado del laboratorio le toman aproximadamente 6 minutos por enfoque, al tomar en cuenta que por muestra se realizan 3 enfoques el tiempo total invertido es de 18min. Es así que, en estas pruebas se realizó la toma de datos en 3 medios de crianza del hongo de género trichoderma por semana.

#### 4.4.1. Primera semana de pruebas

Se prosigue con las pruebas pertenecientes a 21 muestras tomadas en una semana de funcionamiento en el laboratorio de BiosebOrganics. Se debe aclarar que por cada muestra visualizada en el microscopio y con la ayuda de la cámara Neubauer, se toman 3 fotografías en total. Tomando en cuenta las disoluciones se muestra en la Tabla 24 la disolución de 1ml a 10ml y en la Tabla 25 la disolución de 1ml a 10ml.

Tabla 24

*Comparación de conteo automático del sistema y conteo manual a concentración (1/10) ml en pruebas de producción, primera semana*

Número	Conteo manual del número de esporas	Tiempo de conteo manual	Conteo automático del número de esporas	Tiempo de conteo automático	FPVN
1	54	1.30min	53	3seg	-1 2
2	64	1.30min	59	3seg	-5 2
3	64	1.30min	62	3seg	-2 3
4	58	1.30min	53	3seg	-4 2
5	57	1.30min	51	3seg	-6 0
6	64	1.30min	62	3seg	2 7
7	74	2min	70	3.4seg	-4 2
8	74	2min	78	3.4seg	4 2
9	86	2.1min	88	3.4seg	2 3
10	82	2.1min	81	3.4seg	-1 2
11	64	2.1min	62	3.4seg	-2 1

Tabla 25

*Comparación de conteo automático del sistema y conteo manual a concentración (1/100) ml en pruebas de producción, primera semana*

Número	Conteo manual del número de esporas	Tiempo de conteo manual	Conteo automático del número de esporas	Tiempo de conteo automático	FPVN
12	14	17seg	14	3seg	0 0
13	26	25seg	25	3seg	0 4

14	32	28seg	31	3seg	-1 0
15	14	17seg	13	3seg	-1 0
16	17	19seg	16	3seg	-1 0
17	14	17seg	13	3seg	-1 0
18	10	14seg	9	3seg	-1 0
19	9	13seg	8	3seg	-1 0
20	9	13seg	9	3seg	0 0
21	13	16seg	12	3seg	-1 1

#### 4.4.2. Segunda semana de pruebas

Se continúa con las pruebas pertenecientes a 21 muestras tomadas en la segunda semana de funcionamiento en el laboratorio de BiosebOrganics. Tomando en cuenta las disoluciones (explicadas al inicio de esta sección) se observa en la Tabla 26 la disolución de 1ml a 10ml y en la Tabla 27 la disolución de 1ml a 10ml.

Tabla 26

Comparación de conteo automático del sistema y conteo manual a concentración (1/10) ml en pruebas de producción, segunda semana

Número	Conteo manual del número de esporas	Tiempo de conteo manual	Conteo automático del número de esporas	Tiempo de conteo automático	FPVN
1	85	2.20min	84	3.4seg	-1 0
2	63	1.40min	60	3seg	-3 1
3	58	1.30min	59	3seg	-2 3
4	59	1.30min	56	3seg	-1 4
5	63	1.35min	67	3seg	0 4
6	54	1.20min	56	3seg	-1 3
7	69	2min	66	3.4seg	-3 0
8	48	1.28min	47	3.4seg	-2 1
9	39	2.1min	38	3.4seg	-3 2
10	41	1.30min	41	3.4seg	0 0
11	56	1.22min	54	3.4seg	-2 4
12	72	2min	70	3.3seg	-3 1
13	87	2.30min	88	3.4seg	-4 3

Tabla 27

Comparación de conteo automático del sistema y conteo manual a concentración (1/100) ml en pruebas de producción, segunda semana

Número	Conteo manual del número de esporas	Tiempo de conteo manual	Conteo automático del número de esporas	Tiempo de conteo automático	FPVN
14	12	15seg	12	3seg	0 0
15	24	26seg	22	3seg	-2 2

16	13	16seg	12	3seg	-1 0
17	16	19seg	16	3seg	1 2
18	17	21seg	16	3seg	-2 0
19	27	28seg	26	3seg	-1 0
20	16	19seg	15	3seg	-1 3
21	18	18seg	17	3seg	-1 1

#### 4.4.3. Tercera semana de pruebas

Se da continuación con las pruebas pertenecientes a 21 muestras tomadas en la tercera semana de funcionamiento del dispositivo en el laboratorio de BiosebOrganics. Tomando en cuenta las disoluciones (explicadas al inicio de esta sección) la Tabla 28 contiene la disolución de 1ml a 10ml y la Tabla 29 la disolución de 1ml a 10ml.

Tabla 28

*Comparación de conteo automático del sistema y conteo manual a concentración (1/10) ml en pruebas de producción, tercera semana*

Número	Conteo manual del número de esporas	Tiempo de conteo manual	Conteo automático del número de esporas	Tiempo de conteo automático	FPVN
1	75	2min	74	3.4seg	-1 0
2	81	2.10min	82	3seg	-2 1
3	69	1.50min	65	3seg	-4 0
4	58	1.30min	55	3seg	-4 1
5	67	1.35min	66	3seg	-2 1
6	54	1.28min	50	3seg	-5 1
7	51	1.30min	52	3.4seg	-1 0
8	37	1min	35	3.4seg	-3 1
9	46	1.20min	48	3.4seg	-3 1
10	41	1.10min	39	3.4seg	-2 0
11	94	2.50min	94	3.4seg	0 0
12	93	2.48min	90	3.3seg	-3 0
13	105	3min	103	3.6seg	-3 1

Tabla 29

*Comparación de conteo automático del sistema y conteo manual a concentración (1/100) ml en pruebas de producción, tercera semana*

Número	Conteo manual del número de esporas	Tiempo de conteo manual	Conteo automático del número de esporas	Tiempo de conteo automático	FPVN
14	26	26seg	24	3seg	-2 0
15	30	28seg	28	3seg	-3 1
16	11	12seg	11	3seg	-1 1
17	10	13seg	9	3seg	-2 1

18	12	15seg	13	3seg	1 0
19	16	20seg	18	3seg	-1 1
20	19	23seg	20	3seg	1 0
21	28	29seg	27	3seg	-1 0

#### 4.5. Validación y métrica de eficiencia

Con el fin de comprobar la eficiencia del sistema, se usa tablas de comparación (con la ayuda de matrices de confusión) entre los resultados obtenidos usando el sistema frente al conteo manual. Tomando como métrica la denominada “precisión del sistema” que representa el porcentaje total de elementos clasificados correctamente por el sistema de detección y conteo mediante técnicas de visión artificial.

##### 4.5.1. Métrica de eficiencia

Se toma como métrica de precisión del sistema y tasa de error del sistema donde se contabiliza la cantidad de:

- Verdaderos positivos: Número de casos (esporas) en donde el sistema las declara positivas y que son realmente positivos.
- Falsos positivos: Número de casos (esporas) en donde el sistema las declara negativas y que son positivas.
- Verdadero negativo: Número de casos (esporas) en donde el sistema las declara verdaderas y que son falsas.
- Falsos negativos: Número de casos (esporas) en donde el sistema las declara negativas y que son negativas.

##### 4.5.1.2. Matriz de confusión

La matriz de confusión se considera una herramienta para la visualización del desempeño de un algoritmo o sistema usado para el aprendizaje supervisado (en este caso el sistema de visión artificial). Cada columna de esta matriz representa el número de detecciones (detección de esporas de hongo trichoderma), mientras que cada fila representa a las instancias reales (conteo manual de esporas de hongo trichoderma).

Se inicia observando la matriz de confusión de las pruebas de funcionamiento vistas en el punto 4.5 en este mismo capítulo. Donde se analiza los resultados obtenidos y se deduce que estos tendrán un margen de error superior debido a que, solamente en

este caso, las imágenes de esporas de hongo trichoderma no fueron capturadas por un profesional experimentado en manejo de instrumental de laboratorio. En la Tabla 30 se muestra esta primera matriz, donde se hace una sumatoria total de las 20 muestras de esta prueba dentro del formato de matriz de confusión.

#### 4.5.1.3. Métricas de eficiencia pruebas de funcionamiento

Tabla 30

Matriz de confusión de las pruebas de funcionamiento

		Predicción	
		POSITIVO	NEGATIVO
Conteo manual	POSITIVO	VP: 875	FN: -
	NEGATIVO	FP: 56	VN: 35

- **Precisión del sistema en pruebas de funcionamiento**

Se considera a la precisión como el porcentaje total de elementos clasificados correctamente, esta métrica proporciona la probabilidad de que, si ha detectado una espora trichoderma en la prueba, realmente sea una espora trichoderma. En la ecuación 3 se observa la fórmula de la métrica Precisión del sistema, donde se observa que la precisión es de un 93%.

$$\text{Precisión} = \frac{VP}{VP+FP}$$

$$\text{Precisión} = \frac{875}{875+56} \times 100 = 93\%$$

(3)

#### 4.5.1.4. Métricas de eficiencia pruebas de producción

Las matrices de confusión pertenecientes a las pruebas de producción son en total seis debido a que se usan dos diluciones diferentes (de 1ml en 10ml y de 1ml en 100ml), explicadas en el ítem 4.6, además fueron tomadas en tres semanas diferentes.

En la Tabla 31 se observa la matriz perteneciente a la tabla de dilución 1ml a 10ml. En la Tabla 32 se halla la matriz perteneciente a la dilución de 1ml a 100ml, ambas tablas pertenecen a la semana uno de toma de datos.

Tabla 31

Matriz de confusión de las pruebas de producción en disolución de 1ml a 10ml de la primera semana de muestreo

		Predicción	
		POSITIVO	NEGATIVO
Conteo manual	POSITIVO	VP: 693	FN: -
	NEGATIVO	FP: 33	VN: 26

- **Precisión del sistema en pruebas de producción a disolución 1ml en 10ml en la primera semana de almacenamiento de datos**

Es la probabilidad de que, si ha detectado una espora trichoderma en la prueba, realmente sea esta una espora de este género. En la ecuación 4 se observa la fórmula de la métrica Precisión del sistema, donde se observa que la precisión es de un 95%.

$$\text{Precisión} = \frac{VP}{VP+FP}$$

$$\text{Precisión} = \frac{693}{693+33} \times 100 = 95\%$$

(4)

Tabla 32

Matriz de confusión de las pruebas de producción en disolución de 1ml a 100ml de la primera semana de muestreo

		Predicción	
		POSITIVO	NEGATIVO
Conteo manual	POSITIVO	VP: 145	FN: -
	NEGATIVO	FP: 7	VN: 5

- **Precisión del sistema en pruebas de producción a disolución 1ml en 100ml en la primera semana de almacenamiento de datos**

En la ecuación 5 se observa la fórmula de la métrica Precisión del sistema, donde se calcula que la precisión es de un 95% para este caso.

$$\text{Precisión} = \frac{VP}{VP+FP}$$

$$\text{Precisión} = \frac{145}{145+7} \times 100 = 95\%$$

(5)

En la Tabla 33 se observa la matriz perteneciente a la tabla de dilución 1ml a 10ml de la segunda semana de toma de datos. A continuación, en la Tabla 34 se encuentra la matriz perteneciente a la dilución 1ml a 100ml de la misma semana.

Tabla 33

Matriz de confusión de las pruebas de producción en disolución de 1ml a 10ml de la segunda semana de muestreo

		Predicción	
		POSITIVO	NEGATIVO
Conteo manual	POSITIVO	VP: 760	FN: -
	NEGATIVO	FP: 25	VN: 26

- **Precisión del sistema en pruebas de producción a disolución 1ml en 10ml en la segunda semana de almacenamiento de datos**

En la ecuación 6 se observa la fórmula de la métrica Precisión del sistema, donde se calcula que la precisión es de un 96% para este caso.

$$\text{Precisión} = \frac{VP}{VP+FP}$$

$$\text{Precisión} = \frac{760}{760+25} \times 100 = 96\%$$

(6)

Tabla 34

Matriz de confusión de las pruebas de producción en disolución de 1ml a 100ml de la segunda semana de muestreo

		Predicción	
		POSITIVO	NEGATIVO
Conteo manual	POSITIVO	VP: 128	FN: -
	NEGATIVO	FP: 9	VN: 8

- **Precisión del sistema en pruebas de producción a disolución 1ml en 10ml en la segunda semana de almacenamiento de datos**

En la ecuación 7 se observa la fórmula de la métrica Precisión del sistema, donde se calcula que la precisión es de un 93% para este caso.

$$\text{Precisión1} = \frac{VP}{VP+FP}$$

$$\text{Precisión1} = \frac{128}{128+9} \times 100 = 93\%$$

(7)

La Tabla 35 contiene la matriz perteneciente a la dilución 1ml a 10ml de la tercera semana de tomado de datos. En la Tabla 36 se encuentra la matriz correspondiente a la dilución 1ml a 100ml en la misma semana.

Tabla 35

Matriz de confusión de las pruebas de producción en disolución de 1ml a 10ml de la tercera semana de muestreo

		Predicción	
		POSITIVO	NEGATIVO
Conteo manual	POSITIVO	VP: 853	FN: -
	NEGATIVO	FP: 33	VN: 7

- **Precisión del sistema en pruebas de producción a disolución 1ml en 10ml en la tercera semana de almacenamiento de datos**

Es la probabilidad de que, si ha detectado una espora trichoderma en la prueba, realmente sea esta una espora de este género. En la ecuación 8 se observa la fórmula de la métrica Precisión del sistema, donde se observa que la precisión es de un 95%.

$$\text{Precisión1} = \frac{VP}{VP+FP}$$

$$\text{Precisión1} = \frac{853}{853+33} \times 100 = 96\%$$

(8)

Tabla 36

Matriz de confusión de las pruebas de producción en disolución de 1ml a 100ml de la tercera semana de muestreo

		Predicción	
		POSITIVO	NEGATIVO
Conteo manual	POSITIVO	VP: 150	FN: -
	NEGATIVO	FP: 8	VN: 4

- **Precisión del sistema en pruebas de producción a disolución 1ml en 100ml en la primera semana de almacenamiento de datos**

En la ecuación 9 se observa la fórmula de la métrica Precisión del sistema, donde se calcula que la precisión es de un 95% para este caso.

$$\text{Precisión1} = \frac{VP}{VP+FP}$$

$$\text{Precisión1} = \frac{150}{150+8} \times 100 = 95\%$$

(9)

#### 4.5.1.5. Promedio de la métrica precisión del sistema

En la Tabla 37 se aprecia el promedio de todos los datos obtenidos gracias a la matriz de confusión y sus respectivas métricas. Hay que resaltar que este proceso toma en cuenta las pruebas de producción más no las pruebas de funcionamiento locales.

Tabla 37

Promedio de métricas del sistema

Muestreo	Precisión del sistema
Pruebas de producción, primera semana (1/100) ml	95%
Pruebas de producción, primera semana (1/10) ml	95%
Pruebas de producción, segunda semana (1/100) ml	96%
Pruebas de producción, segunda semana (1/10) ml	93%
Pruebas de producción, tercera semana (1/100) ml	96%
Pruebas de producción, tercera semana (1/10) ml	95%
<b>Promedio de pruebas de producción</b>	<b>95%</b>

#### 4.6. Costo/Beneficio

La Tabla 38 contiene el listado de componentes de hardware necesarios para el sistema de detección y conteo de esporas trichoderma, además de la cantidad requerida, el nombre del componente, observaciones y su valor tanto unitario como el total. Al final se coloca la sumatoria de todos estos valores. Cabe recalcar que el signo numera len la Tabla 38 representa el número de componentes necesarios para el funcionamiento del sistema.

Tabla 38

Costos de hardware asociados al desarrollo del prototipo

#	Componentes de hardware	Observaciones	Costo unitario	Costo total
1	Placa embebida Jetson Nano	Tomar en cuenta que el precio aumenta por su importación a Ecuador	\$ 180	\$ 180
1	OMAX cámara USB, de 5Mpx con acople a microscopio	Cámara especializada para observación en microscopio	300	300
1	Microscopio	Componente inventariado en la empresa BiosebOrganics	0	0
1	Fuente de alimentación	Específica para placa embebida	4	4
1	Protección para placa embebida	Creada en una impresora 3D	20	20
150	Muestras de la mezcla con hongo trichoderma	Componente inventariado en la empresa BiosebOrganics	0	0
1	Laptop o pantalla con entrada HDMI	Estas son utilizadas con el sistema dependiendo de la organización del laboratorio. Componente inventariado en la empresa BiosebOrganics	0	0
1	Conexión a internet mediante tarjeta USB WiFi		20	20
1	Cable HDMI		15	15
<b>TOTAL</b>			<b>\$ 539</b>	<b>\$ 539</b>

Con el objetivo de lograr un libre desarrollo del software del sistema se optó por opciones sin pago de licencia; sin embargo, se debe señalar que a ciertos softwares se añade una observación debido a que podrían representar un valor futuro o al superar ciertos límites establecidos. En la Tabla 39 se muestra el nombre del software utilizado, su costo y observaciones importantes referentes al mismo.

Tabla 39

Costos de software asociados al desarrollo del prototipo

#	Software	Observación	Costo
1	Visualizador de imágenes para la cámara OMAX		\$0
2	ToupView Python		0
3	Redes neuronales convolucionales (CNN)		0
4	Jupyter Netbook IDE		0
5	Google Drive	En el caso de que la cuenta utilizada para guardar la base de datos de imágenes y etiquetado de estas exceda las 15Gb de almacenamiento, se debe pagar para seguir almacenando más archivos.	0
6	Google Colab	En el caso de que se requiera un reentrenamiento para mejorar la eficiencia del sistema, es necesario revisar si la plataforma “Google Colab” sigue brindando un buen poder de procesamiento libre.	0
7	Plataforma IoT “Ting Speak”	La limitación de esta versión gratuita es el envío de máximo 3 millones de datos al año, siendo un total de 1092 datos enviados por año con una expectativa de que estos se dupliquen en 2023.	0
<b>TOTAL</b>			<b>\$0</b>

La Tabla 40 muestra el costo del trabajo de ingeniería en el diseño del hardware siendo esta principalmente la creación de la caja protectora de la placa embebida. La integración de los componentes incluye la compatibilidad entre estos y la sección de programación toma en cuenta los códigos escritos para los procesos de entrenamiento y detección/conteo del sistema.

Tabla 40

Costos de ingeniería

#	Descripción	Observación	Cantidad	Costo por hora	Costo Total
1	Diseño del hardware	Según el ministerio de trabajo en Ecuador (Ministerio del Trabajo del Ecuador, 2021), el salario mínimo legal de un elaborador de moldes es \$425.	3 horas	2.66	\$7.98
2	Integración de componentes	Según el ministerio de trabajo en Ecuador (Ministerio del Trabajo del Ecuador, 2021), el salario mínimo legal de ingeniero electrónico es \$430.60.	4 horas	2.69	10.77
3	Programación	Según el ministerio de trabajo en Ecuador (Ministerio del Trabajo del Ecuador, 2021), el salario mínimo legal de un programador junior de software es \$426.20; se toma en cuenta que un programador de algoritmo escribe 20 líneas de código al día y por último tanto el código de detección y conteo como el código de entrenamiento consta de 170 líneas.	68 horas	2.66	181.13
<b>TOTAL</b>			<b>75 horas</b>		<b>\$199.88</b>

Los costos de ingeniería (observados en la Tabla 40) son asumidos por el autor del presente proyecto por lo tanto su valor no se contabiliza en la inversión realizada por la empresa privada.

La Tabla 41 es una compilación de los costos observados en las tres tablas anteriores, es necesario aclarar que en esta tabla se incluyen costos de ingeniería debido a que el objetivo de esta es recopilar el costo total del sistema.

*Tabla 41*

*Costos totales*

<b>Ítem</b>	<b>Costo</b>
Hardware	\$539
Software	\$0
Referentes a la ingeniería	\$199.88
<b>TOTAL</b>	<b>\$738.88</b>

Para finalizar se relaciona los costos y beneficios del sistema, teniendo en cuenta que el presente proyecto tuvo un valor de \$738.88 (Tabla 41), los beneficios económicos son los siguientes:

- Mejorar el porcentaje de acierto en el cálculo de calidad del producto, debido a que se utilizaba un conteo manual en una disolución de 1ml de líquido con esporas de hongo trichoderma en 100ml de líquido esterilizado y el sistema de detección y conteo permite realizar esta acción en disoluciones de 1ml de líquido con esporas de hongo trichoderma en 10ml.
- Reducir el tiempo en el conteo de esporas en cada muestra, este punto es clave ya que se contabiliza el tiempo ahorrado y se relaciona con el valor monetario hora/trabajo de los laboratoristas encargados del proceso (siendo este valor \$5). El promedio de tiempo ahorrado es 108.36 segundos, tomado de las muestras del ítem 4.4 de pruebas de producción. Teniendo en cuenta que en una semana se

toman 21 muestras existe un ahorro de 0.63 horas en este periodo de tiempo, dando un total de 32.7 horas ahorradas al año.

Tabla 42

Tabla Costo/Beneficio

<b>Costo total del proyecto</b>	<b>Tiempo ahorrado al año</b>	<b>Valor hora de trabajo</b>	<b>Tiempo retorno de inversión</b>
\$738.88	32.7 horas	\$5	<b>4.52 años</b>

Si bien la inversión realizada en el presente proyecto tiene un tiempo de recuperación de 4.52 años, como se observa en la Tabla 42, no se ha tomado en cuenta la mejora en cuanto a la etapa de revisión de calidad en la crianza de hongos de género trichoderma debido a que esto no representa un aumento en el precio final del fungicida o una ganancia económica directa. Este fenómeno se da debido a que el prototipo del proyecto permite hacer una revisión en una difusión menor (1ml de líquido con esporas trichoderma en 10ml de líquido estéril) que la ocupada en el conteo manual de esporas menor (1ml de líquido con esporas trichoderma en 100ml de líquido estéril), dando así una menor dispersión de este microorganismo.

Igualmente, el presente sistema abre las puertas para a una mayor producción o cultivo de hongos de género trichoderma, debido a que se agiliza la etapa de control de calidad de este además de hacerla más precisa.

## Conclusiones

La recopilación del estado del arte, especialmente lo referido a la ciencia detrás del cultivo de hongos de género trichoderma, es sumamente necesaria debido a que brinda información nueva y necesaria para el desarrollo del proyecto como las técnicas de control de calidad del producto, técnicas de microscopía más utilizadas entre otras.

El presente proyecto propuso un sistema embebido que corre un algoritmo mediante técnicas de visión artificial con el objetivo de identificar y contabilizar esporas del hongo de género trichoderma, obtenidas con la colaboración de la empresa Bioseb Organics. La placa utilizada está diseñada para trabajar en el campo de machine learning por sus características como una tarjeta gráfica, siendo esta la GPU NVIDIA Maxwell de 128 núcleos, un procesador con un CPU ARM A57 de 4 núcleos, una memoria RAM de 4 GB de LPDDR4 de 64 bits y compartimiento para tarjeta MicroSD.

Se estableció un máximo de 150 muestras para los procesos de entrenamiento del sistema ya que en cada muestra existe un promedio de 61 esporas, teniendo aproximadamente 9150 esporas trichoderma etiquetadas en la base de datos entrenada, considerando que no se debe tener más de ese valor de etiquetas porque se puede caer en sobrealimentación del sistema, de acuerdo con estudios previos.

Para la detección de esporas del hongo trichoderma se utilizó el sistema de detección de objetos Tiny Yolo por su peculiaridad de revisar cada matriz que este crea una sola vez, dando como resultado un algoritmo con un menor consumo de recursos computacionales en relación con otros similares. Cabe recalcar que Tiny Yolo igualmente tiene resultados satisfactorios en la detección de objetos.

Utilizar las técnicas de aprendizaje por herencia es fundamental para el funcionamiento del código en la placa embebida Nvidia Jetson Nano por sus limitantes en cuanto a capacidad computacional, en relación con servidores o computadores y la gran cantidad de recursos necesarios para los procesos de entrenamiento de aprendizaje profundo.

Las pruebas funcionales locales arrojaron un porcentaje del 93% frente al 96% de las pruebas de producción, esto debido a que estas últimas se realizaron por personas con experiencia en el manejo de instrumentos de laboratorio, por lo que se logra una mayor resolución al tener un mejor enfoque en el microscopio.

La inversión tiene un tiempo de recuperación de 4.52 años de utilización del sistema, sin embargo, no se toma en cuenta la mejora en la revisión de calidad en la crianza de hongos de género trichoderma debido a que esto no representa un aumento en el precio del fungicida, pero si existen beneficios en cuanto a la clasificación haciéndola más precisa otorgando al producto final un adicional comercial. Esto se logra gracias a que el sistema permite el uso de una dilución menor en comparación a la anteriormente usada, por lo tanto, las muestras líquidas de esporas trichoderma tienen una menor dispersión en el líquido esterilizado, siendo esta relación 1ml/10ml respectivamente (en el conteo manual anterior a la implementación del prototipo se tenía una relación de 1ml/100ml).

## Recomendaciones

Para una mejor visualización de microorganismos (en este caso de hongos de género trichoderma) tanto en calidad como en ahorro de tiempo se aconseja que el microscopio sea utilizado por personal capacitado, experimentado y en labor.

Para la creación del case de protección se recomienda hacerlo en una impresora 3D para conservar la adaptabilidad con la placa Nvidia Jetson Nano, como lo muestra el diseño especificado en el ítem 3.5.1.3 referente a protección contra manipulación. Además de utilizar el material PLA (ácido poli-láctico) por sus características biodegradables, alta resistencia a flexión (4000 a 5000 MPa) y tracción (50 a 70 MPa).

Para aumentar el margen de aciertos del sistema de detección y conteo de esporas de hongo trichoderma se recomienda utilizar una cámara de mayor resolución, pero con las mismas características de acople y compatibilidad a microscopio, además de usar técnicas de microscopía de luz transmitida (campo claro) o similares.

En casos de detección de microorganismos con técnicas de visión artificial supervisadas, el etiquetado del objeto en el proceso de entrenamiento conlleva mucho tiempo, por motivo de que en una muestra (imagen) se puede hallar una gran cantidad de microorganismos, por lo que se recomienda asignar un tiempo prudente a este proceso y no superar las 150 muestras.

Para los procesos de entrenamiento, mismos que consumen bastante recurso computacional, no se recomienda usar la placa embebida Nvidia Jetson Nano sino un sistema alternativo, en el caso del presente proyecto se utilizó el servicio en nube “Google Colab”.

Al obtener un bajo porcentaje de error al comparar la disolución de 1ml de líquido con alta concentración de esporas de hongo trichoderma en 10ml de líquido estéril y 1ml de líquido con alta concentración de esporas de hongo trichoderma en 100ml de líquido estéril, se recomienda usar la primera disolución ya que brinda una mejor apreciación de la cantidad de hongos criados.

## ANEXOS:

### ANEXO A: Carta de colaboración de empresa BiosebOrganics



Ibarra, 17 de diciembre del 2021

**Dirigido a:**  
**Universidad Técnica del Norte.**

Asunto: Carta de Compromiso

**Ing. Daniel Jaramillo**

**Coordinador de la carrera CIERCOM-CITEL**

Un cordial saludo a la Universidad Técnica del Norte de parte de la empresa BiosebOrganics Cia. Ltda. de nuestra consideración:

Por la presente, Yo **Maria Fernanda Rodríguez Meza** en calidad de Representante legal de **Bioseborganics Cía. Ltda.**, empresa dedicada a la elaboración de bioproductos basados en microorganismos orientada a la agricultura, nos comprometo a apoyar al estudiante **Noboa Delgado Esteban Marcelo** con cédula de identidad número 100386975-5, para la realización del proyecto **"Sistema embebido para el reconocimiento y conteo de esporas de hongo trichoderma a través de técnicas de visión artificial"** a ser desarrollado con las siguientes condiciones:

- BiosebOrganics se compromete al auspicio concerniente al costo, información perenne y acceso a instalaciones para el desarrollo del proyecto de titulación.
- Se llevará seguimiento al trabajo efectuado con el fin de presentar una carta de conformidad al finalizar el proyecto.
- Aceptamos que la propiedad intelectual es la de Universidad Técnica del Norte (UTN), sin embargo, el trabajo puntual podrá ser utilizado por nuestra empresa sin requerimiento de autorización alguna.

Por la atención a lo presente anticipo mi agradecimiento

**ATENTAMENTE**



**Ing. María Fernanda Rodríguez Meza**  
**Gerente General**  
**Bioseborganics Cía. Ltda.**

## **ANEXO B: Formato de cuestionario con resultados**

### **CUESTIONARIO DIRIJIDO AL ING. GABRIEL CURIPALLO**

El objetivo de este cuestionario es recopilar información de interés referente al proceso de crianza del hongo de género trichoderma enfocándose en la etapa de revisión de calidad de resultados. Específicamente qué materiales ya posee la empresa que se puedan utilizar en el sistema, requisitos de la forma del dispositivo tecnológico, métodos de control de calidad empleados, técnicas de control de calidad empleadas, técnicas de conteo de esporas, cualidades visibles a través del microscopio para esporas de hongo trichoderma, como diferenciar las esporas de hongo trichoderma de microorganismos invasores, qué clase de organismos invasores han sido detectados, qué hacen cuando hay microorganismos intrusos y el tiempo promedio utilizado en el conteo de esporas (proceso de control de calidad).

La participación en este estudio es completamente voluntaria, sin embargo, se puedo omitir preguntas en el caso de no poder divulgar cierta información. Las respuestas brindadas del estudio serán utilizadas exclusivamente para el presente proyecto y en los repositorios de la Universidad Técnicas del Norte.

Por su tiempo y apoyo le estamos agradecidos.

**1. ¿Qué materiales que la empresa posee y se pueda usar en el presente proyecto?**

- a) Microscopio
- b) Microscopio digital
- c) Cámara con acople a microscopio con soporte para Linux
- d) Placa embebida
- e) Pantalla
- f) Maus y teclado
- g) Otro: \_\_\_\_\_

**2. ¿Qué forma sería más cómoda de usar en el sistema?**

- a) Tamaño aproximado a un computador de escritorio
- b) Tamaño aproximado a una laptop (computador portátil)

c) Tamaño pequeño (equiparable a un mini PC o NUC)

**3. ¿Qué métodos de control de calidad usan?**

**4. ¿Qué técnica de conteo de esporas utilizan?**

**5. ¿Cuáles son las cualidades visibles, microscópicamente, de la spora de hongo trichoderma?**

**6. ¿Cómo diferenciar las esporas de hongo trichoderma en relación con posibles microorganismos invasores?**

**7. ¿Qué microorganismos invasores se han detectado?**

**8. ¿Qué procedimiento se realiza una vez hecho el conteo de esporas de hongo trichoderma?**

**9. ¿Qué procedimiento realizan al encontrar en el control de calidad cualquier cantidad de microorganismos invasores?**

**10. ¿Qué tiempo promedio demora un técnico en el conteo de esporas de hongo trichoderma en una sola muestra?**

## **CUESTIONARIO RESPONDIDO**

**1. ¿Qué materiales que la empresa posee y se pueda usar en el presente proyecto?**

h) Microscopio

i) Microscopio digital

j) Cámara con acople a microscopio con soporte para Linux

k) Placa embebida

l) Pantalla

m) Maus y teclado

n) Otro: \_\_\_\_\_

**2. ¿Qué forma sería más cómoda de usar en el sistema?**

d) Tamaño aproximado a un computador de escritorio

e) Tamaño aproximado a una laptop (computador portátil)

f) Tamaño pequeño (equiparable a un mini PC o NUC)

**3. ¿Qué métodos de control de calidad usan?**

Determinación de la creación de esporas, determinación de contaminantes y análisis de concentración de esporas, ausencia de bacterias bacillus y cocos.

**4. ¿Qué técnica de conteo de esporas utilizan?**

Conteo manual de esporas con contador mecánico y uso de hemo citómetro o cámara de Neubauer, conteo de cuadrículas en forma diagonal con la ayuda del hemo citómetro y la dilución del compuesto en agua pura de 1 a 100.

**5. ¿Cuáles son las cualidades visibles, microscópicamente, de la espora de hongo trichoderma?**

Forma ovalada y de tamaño mayor en relación a posibles microorganismos invasores

**6. ¿Cómo diferenciar las esporas de hongo trichoderma en relación a posibles microorganismos invasores?**

Mayor tamaño y forma ovalada continúa

**7. ¿Qué microorganismos invasores se han detectado?**

Mayormente bacterias como Bacillus (forma muy alargada), cocos y la unión de cocos (forma circular y la unión de esta figura).

**8. ¿Qué procedimiento se realiza una vez hecho el conteo de esporas de hongo trichoderma?**

Se compara el número de esporas en la muestra con una tabla de estas para comprobar la cantidad de hongo producido en el birreactor mediante el proceso de crianza

**9. ¿Qué procedimiento realizan al encontrar en el control de calidad cualquier cantidad de microorganismos invasores?**

En ese caso se descarta toda la mezcla del birreactor

**10. ¿Qué tiempo promedio demora un técnico en el conteo de esporas de hongo trichoderma en una sola muestra?**

25 min

**Firma de autorización**

<p>Ing. Fabián Geovanny Cuzme Rodríguez</p>  <p>_____ Director</p>	<p>Sr. Esteban Marcelo Noboa Delgado</p>  <p>_____ Estudiante</p>	<p>Ing. Gabriel Curipallo</p>  <p>_____ Encargado laboratorio</p>
---	--	--

## ANEXO C: Revisión de trabajos relacionados

### Objetivos:

- De cada trabajo extraer información relevante para el estudio
- Sacar los requisitos de esas partes

El presente documento sintetiza la revisión de trabajos o investigaciones relacionadas al tema del proyecto de grado, con el fin de establecer requerimientos técnicos tecnológicos funcionales para el diseño y la implementación del sistema contador de esporas de hongos del género trichoderma mediante técnicas de visión artificial.

### 1. Trabajo relacionado 1:

Ficha bibliográfica 1			
Autor:	Naranjo y Silva	Editorial:	Universidad de las Fuerzas Armadas
Título:	“Diseño e implementación de un sistema para la contabilización automática de unidades formadoras de colonias bacterianas: coliformes, mesófilos aerobios, bacilos, estafilococos y levaduras usando técnicas de visión artificial”		
Año:	2016	Ciudad:	Quito

#### Resumen del contenido:

- **Visión artificial: Labview**

Debido a que esta plataforma de desarrollo de sistemas gráfico cuenta con un módulo llamado Labview NI Vision Development, donde se hallan varios algoritmos y funciones de visión artificial enfocados al procesamiento y mejoramiento de imágenes.

- **Cámara fotográfica: Nikon modelo D90**

Esta cámara posee una resolución de 12.3 megapíxeles. Sin embargo, no posee un acople para microscopio debido a que este proyecto crea su propia armazón para sujetar a la cámara y no utiliza un microscopio para la observación de sus colonias bacterianas.

- **Adquisición de la imagen: digiCamControl versión 2.0**

Este es un software para controlar los valores de tiempo de exposición, apertura de diafragma, enfoque y adquisición de fotografía en una carpeta del ordenador de la cámara Nikon.

- **Tratamiento de la imagen: Herramienta ROI**

ROI es una herramienta que tiene por objetivo denotar una zona o región de interés. Además, usa una técnica, no mencionada, para convertir una imagen con color (imagen RGB) a una imagen a escala de grises.

## 2. Trabajo relacionado 2:

---

Ficha bibliográfica 2			
Autor:	Álvarez Ramos, Nino, & Santos	Editorial:	IEEE
Título:	“Automatic classification of Nosema pathogenic agents through machine vision techniques and kernel-based vector machines”		
Año:	2013	Ciudad:	Quito

---

### Resumen del contenido:

- **Adquisición de imagen: Microscopio fluorescente electrónico con lentes de 40x y 100x**

Este microscopio es usado para la adquisición de imágenes para su posterior clasificación automática mediante técnicas de visión artificial, las imágenes para la matriz de aprendizaje son donadas por el Hospital de la facultad de veterinaria de la Universidad de Complutense Madrid.

- **Segmentación**

Proceso para delimitar dentro de una imagen los objetos de interés creando subdivisiones o segmentos para un objeto a ser detectado, en esta zona se citan dos técnicas donde la primera se basa en detección de valores umbrales de la imagen y la segunda implica la segmentación de la imagen en función de cambios en los niveles de gris.

- **Extracción de características: Detección externa del espacio de escala a través de la función de Gauss y localización de puntos clave**

Se necesita obtener una buena cantidad de características altamente distintivas del objeto a identificar (en este caso espora de hongo trichoderma).

La detección extrema del espacio de escala es una primera etapa que es utilizada para identificar puntos de interés que son invariantes a la orientación y la escala; mientras que la acción de localización de puntos clave tiene como principal función que los modelos detallados determinen la ubicación y escala de cada zona de potencial interés.

---

## 3. Trabajo relacionado 3

---

Ficha bibliográfica 3			
Autor:	Yordanov & Mladenov	Editorial:	IEEE
Título:	“Humanoid robot detecting animals via neural network”		
Año:	2013	Ciudad:	Belgrade

---

### Resumen del contenido:

- **Red neuronal: Uso de tensor flow y red neuronal convolucional**

TensorFlow es una biblioteca de software de código abierto que se usa en el cálculo numérico mediante gráficos flujo de datos, inicialmente desarrollado por Google Brain Team y es una multiplataforma compatible con el lenguaje de alto nivel Python.

Una red neuronal convolucional es una clase de redes neuronales profundas y de retroalimentación, más comúnmente utilizadas para analizar imágenes visuales, donde esta red se entrena con un conjunto de imágenes que posee 12500 imágenes de perros y 12500 imágenes de gatos.

---

#### 4. Trabajo relacionado 4:

Ficha bibliográfica 4			
Autor:	Militante, Gerardo, & Dionisio	Editorial:	IEEE
Título:	: “Plant leaf detection and disease recognition using deep learning”		
Año:	2019	Ubicación:	Euroasia

##### Resumen del contenido:

- **Adquisición de imagen: Repositorio de Plant Village y lenguaje de programación de alto nivel Python**

El conjunto de datos de imágenes utilizado para entrenar el modelo se adquirió en el repositorio de Plant Village, siendo este un conjunto de datos de aproximadamente 35.000 imágenes con 32 clases diferentes de variedades de plantas y sus enfermedades. Además, se utilizó un script de Python para descargar imágenes de las enfermedades de las plantas desde el repositorio.

- **Preprocesamiento de imágenes: Cambio de escala de color, suavizado y redimensionamiento de imagen**

En este proceso se mejora la imagen a una escala de color adecuada y la suaviza, además estas imágenes en color y redimensionadas se procesan a una resolución de 96x96 para su procesamiento.

- **Técnica de visión artificial (clasificación): Red neuronal convolucional**

Una red neuronal convolucional puede detectar y categorizar, es eficiente en la evaluación de imágenes gráficas y extrae características esenciales a través de su estructura de múltiples capas. Esta tecnología incluye cuatro capas, como son: imagen de entrada, capa convolucional o capa de agrupación, capas completamente conectadas y salida.

#### Firma autorización

<p>Ing. Fabián Geovanny Cuzme Rodríguez</p>  <p>_____ Director</p>	<p>Sr. Esteban Marcelo Noboa Delgado</p>  <p>_____ Estudiante</p>
---	--

## ANEXO D: Mesa de trabajo

### Contenido

<b>Programa: 1 de diciembre .....</b>	<b>122</b>
<b>Bienvenida .....</b>	<b>122</b>
<b>Recopilación de requisitos de ANEXOS B Y C .....</b>	<b>123</b>
<b>Según revisión de trabajos relacionados y la encuesta realizada al Ing. Gabriel: .....</b>	<b>123</b>
<i>Adquisición de datos .....</i>	<i>123</i>
<b>Hardware y software .....</b>	<b>123</b>
<b>Análisis del aumento y recolección de requisitos.....</b>	<b>128</b>
<b>Definición de importancia de cada requisito .....</b>	<b>129</b>
<b>Placa embebida.....</b>	<b>129</b>
<b>Cámara con acople a microscopio .....</b>	<b>130</b>
<b>Protección externa .....</b>	<b>131</b>
<b>Lenguaje de programación .....</b>	<b>131</b>
<b>Plataforma en nube compatible con Edge Computing .....</b>	<b>132</b>

## “Investigación colaborativa entre autor de tesis y director”

Realizar una reunión técnica tecnológica para constatar todos los elementos necesarios, en cuanto a tecnología, del sistema y reunir todos los requisitos obtenidos con las anteriores técnicas de recopilación (encuesta y revisión de trabajos relacionados) definiendo su importancia, poner a revisión otros requisitos necesarios para el sistema y brindar posibles candidatos a utilizarse en cada uno de los ítems colocados.

### 1. Programa: 1 de diciembre

Tabla 43:

*Cronograma de la actividad mesa de trabajo*

3:00 – 3:02	Bienvenida Ing. Fabián Cuzme, director del presente proyecto de grado Sr. Esteban Noboa, estudiante de trabajo de grado
3:02 – 3:30	<ul style="list-style-type: none"><li>• Trabajo en equipo: Lluvia de idea en base al conocimiento del director, la encuesta hacia el Ing. Gabriel y la investigación de trabajos relacionados.</li><li>• Adquisición de datos.</li></ul>
3:30 – 4:00	<ul style="list-style-type: none"><li>• Trabajo en equipo: localización de hardware y software para tablas de comparación y elección de las mejores características.</li></ul>
4:00 – 4:30	<ul style="list-style-type: none"><li>• Trabajo en equipo: selección de importancia de cada requisito para colocar en tablas de comparación.</li></ul>

### **Bienvenida**

Pequeña bienvenida de las personas presentes en la mesa de trabajo: Ing. Fabián Cuzme como director del proyecto de grado y Sr. Esteban Noboa como estudiante responsable del trabajo de grado.

## **Recopilación de requisitos de ANEXOS B Y C**

### **Según revisión de trabajos relacionados y la encuesta realizada al Ing. Gabriel:**

En este ítem se desglosa los componentes utilizados en cada uno de los trabajos relacionados de alto impacto y se los relaciona con los requisitos obtenidos mediante la encuesta. En la Figura 42 se observa las anotaciones principales del ejercicio que posteriormente se tomarán como base para la creación de requisitos de diferentes niveles.

#### ***Adquisición de datos***

En este ítem se detallan requisitos adicionales a los ya encontrados gracias a los métodos de adquisición de datos encuesta y revisión de trabajos relacionados, además se colocan descripciones de características que deben cumplir los requisitos vistos entre el autor y el director en la mesa de trabajo correspondiente al presente anexo.

#### **Hardware y software**

Posibles elementos hardware o aplicaciones software compatibles con las características de los requisitos, ya especificadas, con el fin de usarlas como una referencia al momento de aplicar tablas de comparación.

Así, se observa en la Figura 49 un mapa mental de los 5 requisitos generales que se colocó. En la Figura 50 se aprecia los elementos hardware con sus respectivos requisitos y candidatos para su selección en este caso de cámara digital y placa embebida, en la Figura 50 se encuentran las aplicaciones software enfocadas en el tratamiento de la imagen y el lenguaje de programación donde igualmente se colocan sus respectivos requisitos y candidatos (cabe recalcar que se decidió antes de este espacio de trabajo que se utilizará el lenguaje de alto nivel Python). En la Figura 52 existe un cuadro conceptual donde se encuentran las técnicas de visión basadas en inteligencia artificial con sus respectivos requisitos y candidatos.

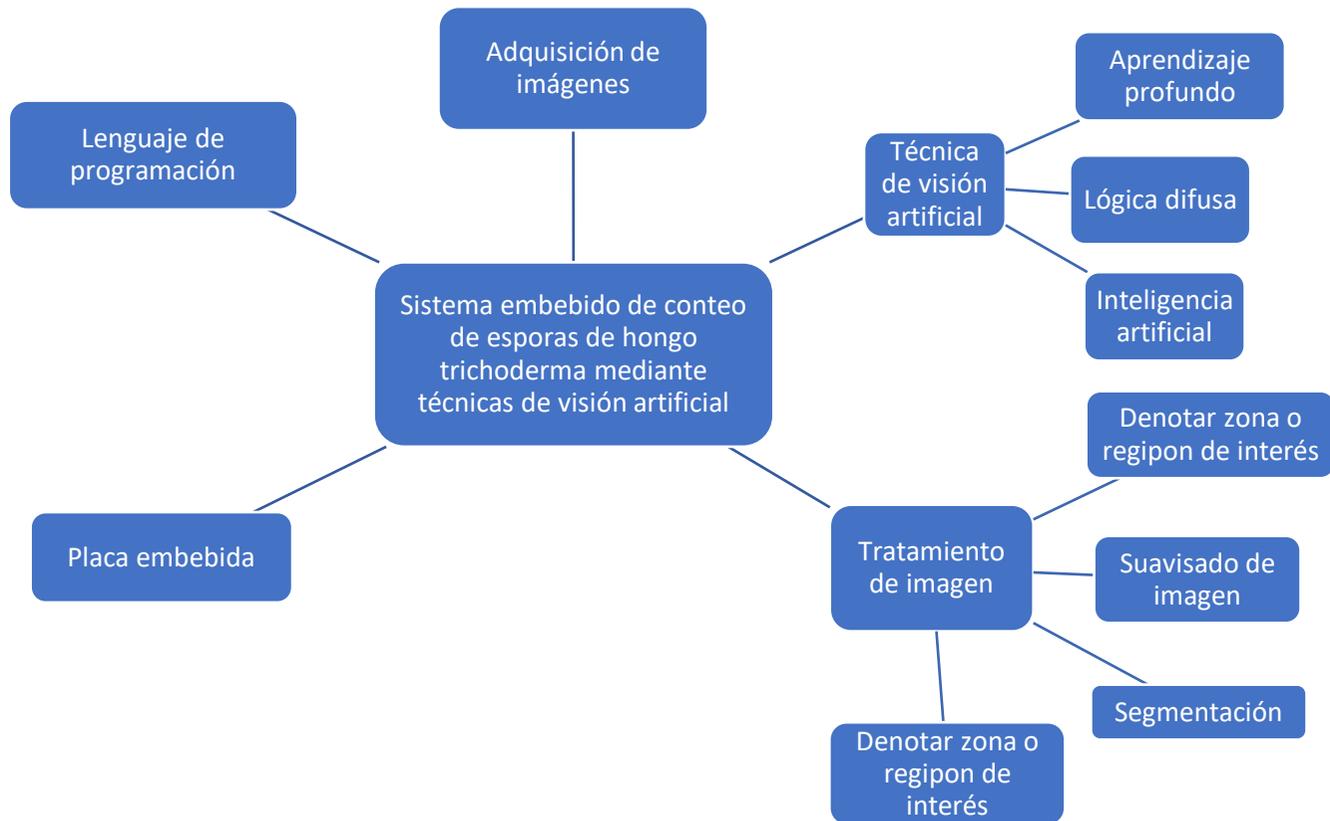


Figura 49:

Mapa resultante de lluvia de ideas, ideas principales

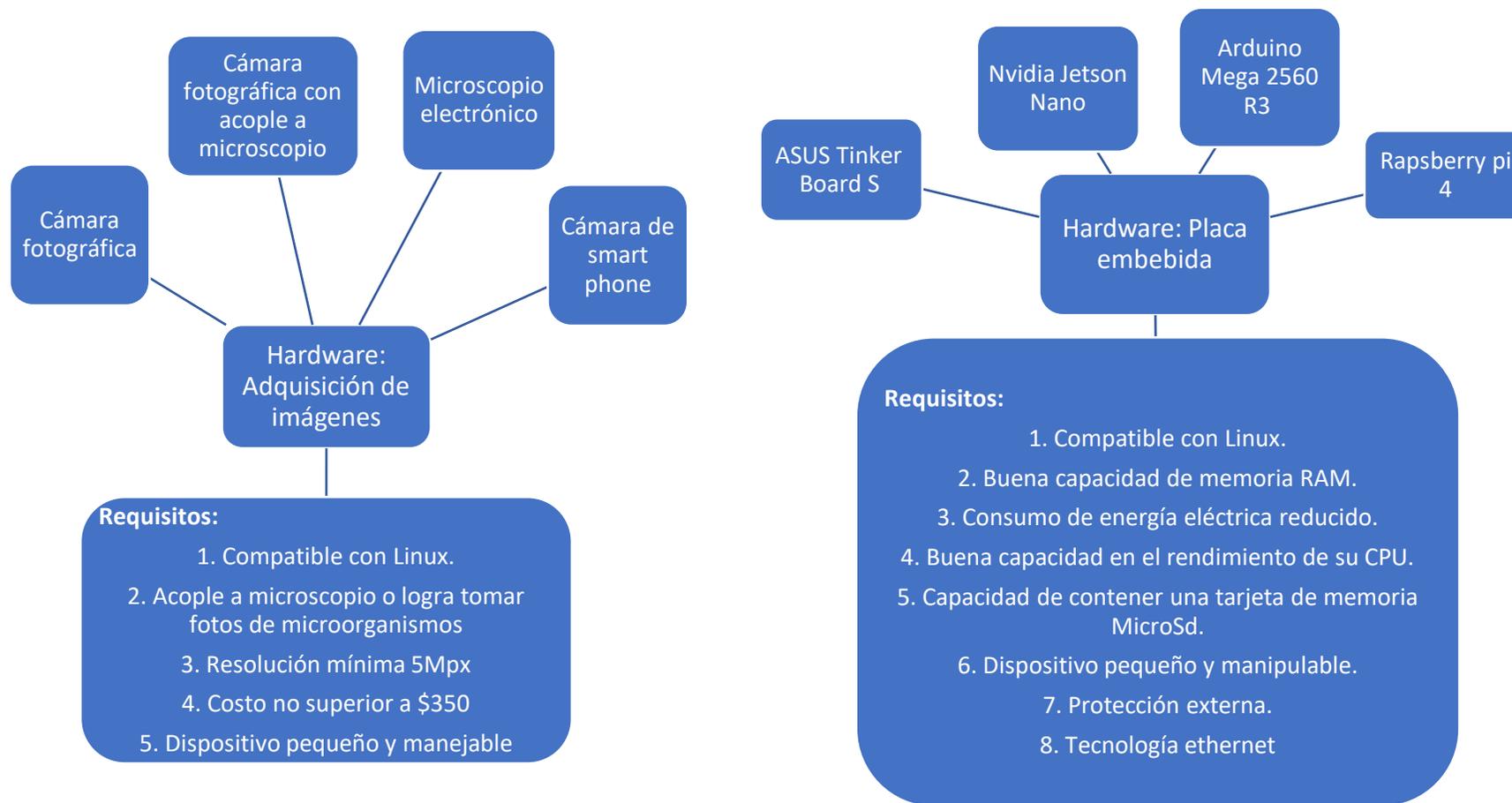


Figura 50:

Hardware para adquisición de imágenes y placas embebidas

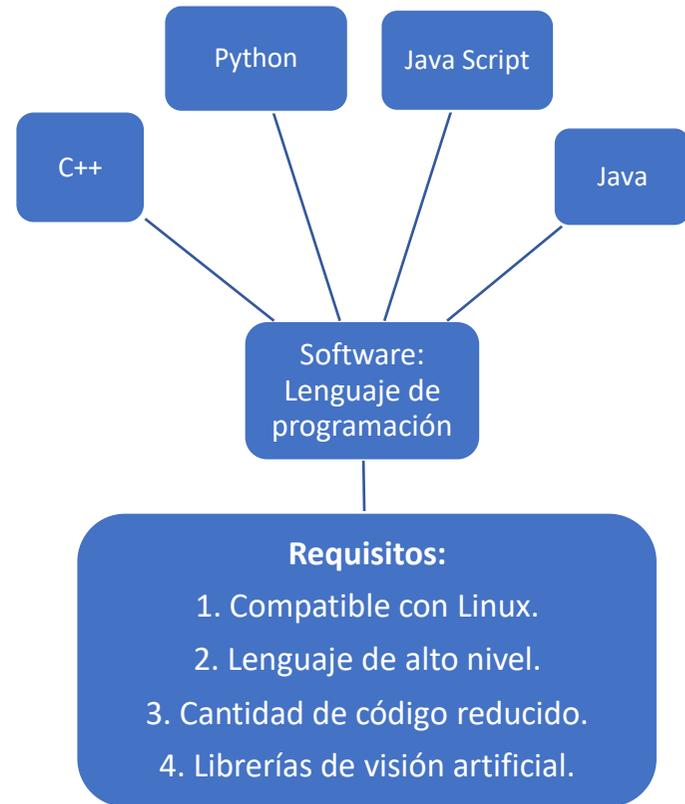
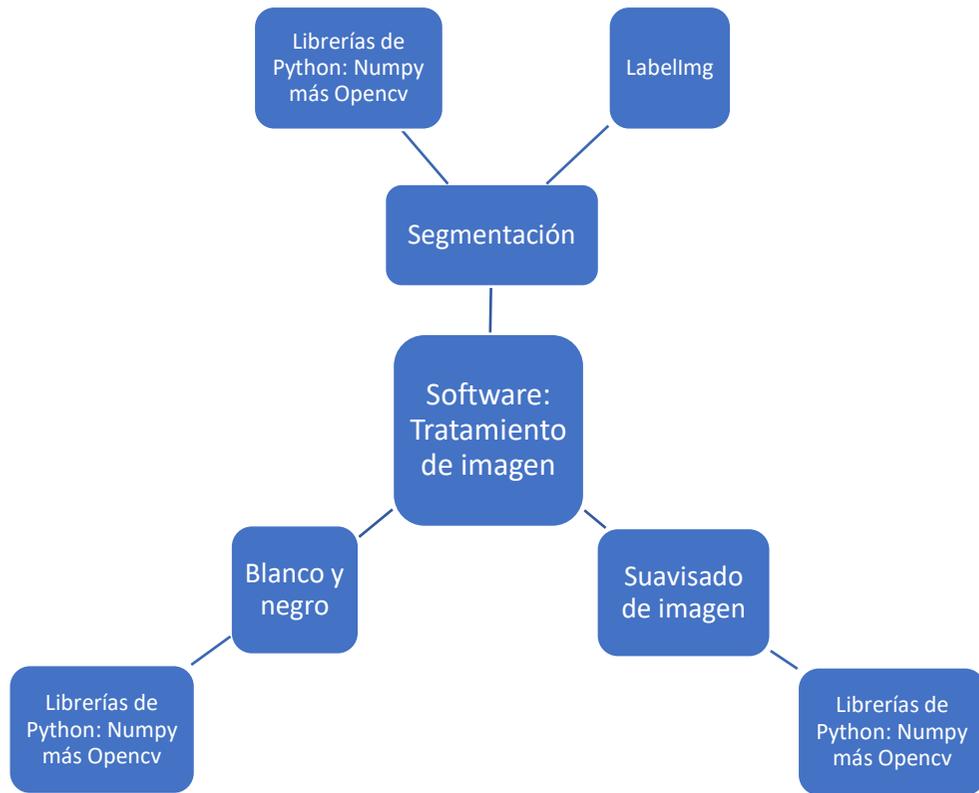


Figura 51:

Software para tratamiento de imágenes y lenguaje de programación

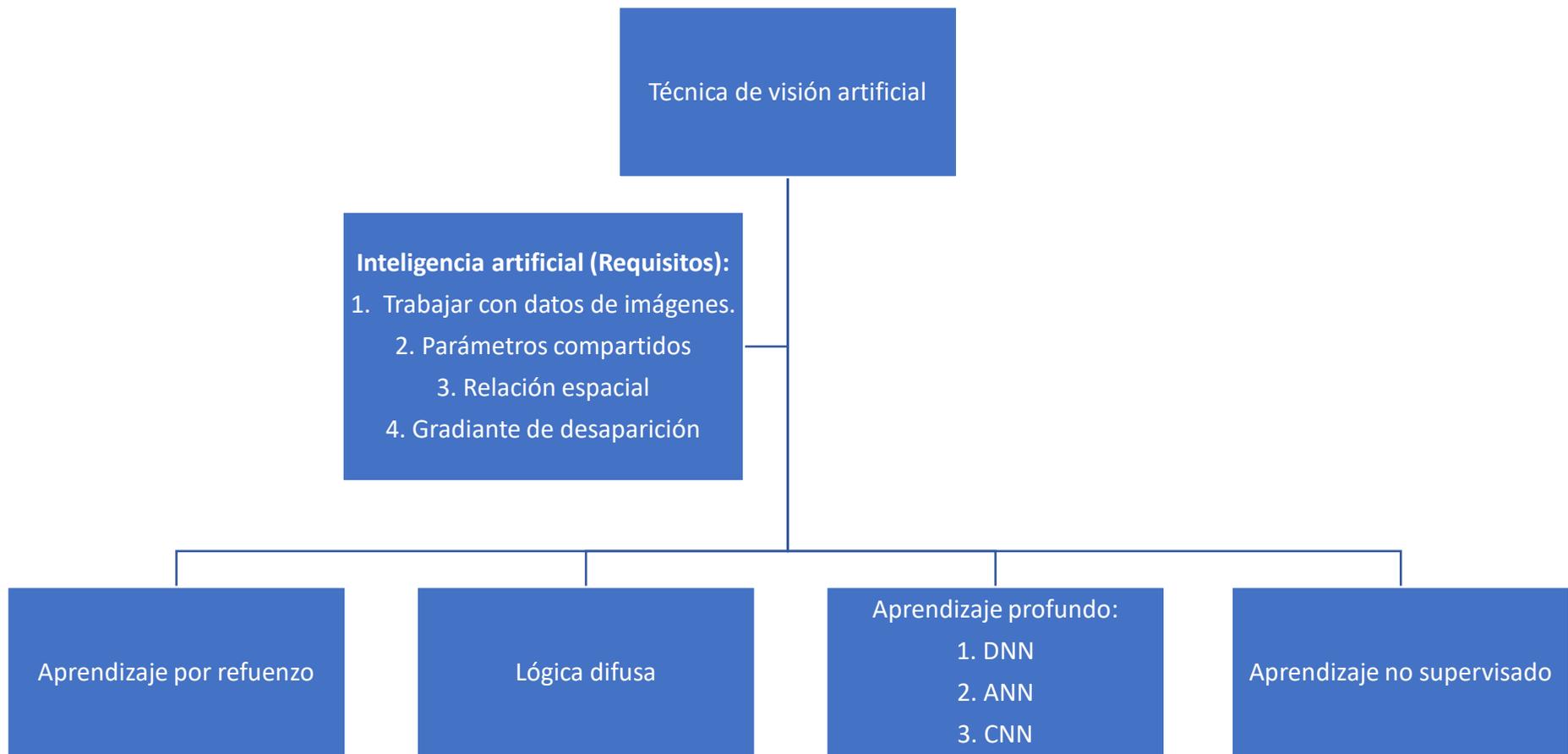


Figura 52:

Técnicas de visión artificial

*i. Análisis del aumento y recolección de requisitos*

En la Figura 50 se observan 5 requisitos generales, que posteriormente se irán desglosando, en lo referente a hardware: Placa embebida y Adquisición de imagen, en cuanto a software se encuentra: Lenguaje de programación, tratamiento de imagen y técnicas de visión artificial.

En la Figura 51 se muestra la parte de hardware del sistema donde ya se han definido requisitos y candidatos para los ítems: Adquisición de imagen y placa embebida.

- En cuanto a adquisición de imagen los requisitos son: compatibilidad con Linux, capacidad de acople a microscopio o logra tomar fotos de microorganismos (microscopio digital), resolución mínima 5Mpx y costo no superior a \$350. Además de posibles dispositivos como para estos requisitos como son: cámara fotográfica, cámara fotográfica con acople a microscopio, microscopio electrónico y Smart Phone.
- En cuanto a la Placa embebida se citan sus requisitos: compatibilidad con Linux, Buena capacidad de memoria RAM, consumo de energía eléctrica reducido, buena capacidad en el rendimiento de su CPU, capacidad de contener una tarjeta de memoria MicroSD. Además de posibles dispositivos como para estos requisitos como son las placas embebidas: ASUS Tinker Board, Nvidia Jetson Nano, Arduino Mega 2560, RaspBerry pi 4.

En la Figura 52 se observa la parte de software del sistema específicamente en tratamiento de imagen y lenguaje de programación:

- En tratamiento de imagen se colocan varios procesos como suavizado de imagen, segmentación de imagen y escala blanco y negro de imagen, esto se realiza con librerías compatibles con el lenguaje de programación Python que son: Numpy, Opencv y LabelImg.
- En la sección de lenguaje de programación ya se escogió al lenguaje de programación de alto nivel Python, debido a que cumple con todos los requisitos planteados como son: compatibilidad con Linux, lenguaje de alto nivel, cantidad de código reducido y cuenta con librerías de visión artificial.

En la Figura 53 se constata el requisito que investiga técnicas de visión artificial empleado inteligencia artificial, brindando principalmente las opciones que nos brinda el

“aprendizaje profundo” como son: Redes neuronales artificial (ANN), Redes neuronales convolucionales (CNN), Redes neuronales recurrentes (RNN).

*ii. Definición de la importancia de cada requisito*

En el presente ítem se analiza la importancia de cada característica de todos los requisitos coleccionados en este anexo, se cuenta con tres etiquetas las cuales son: alta, media y baja los cuales se detallan en la Tabla 44.

Tabla 44

Especificaciones para tabla de requerimientos

<b>ESPECIF.</b>	<b>DESCRIPCIÓN</b>
Alta	Requerimiento crítico que debe ser considerado en el desarrollo del sistema.
Media	Requerimientos que pueden ser omitidos solo en condiciones de fuerza mayor.
Baja	Requerimientos que se pueden omitir, este requerimiento no genera impacto significativo en el sistema.

Los requisitos tienen una abreviación y numeración vistas en el ítem 3.4.1.3 del capítulo 3 del documento de trabajo de grado, seguido de una descripción acerca de la importancia de este en relación con el sistema y finalmente a qué marca pertenece.

**Placa embebida**

En la siguiente Tabla 45 se puede observar el análisis de la importancia de los requisitos ya definidos, que se toman en cuenta para la elección de la placa embebida para el sistema de conteo de esporas mediante técnicas de visión artificial.

Tabla 45

Análisis requisitos para placa embebida

<b>Requisito</b>	<b>Análisis</b>	<b>Alta</b>	<b>Med</b>	<b>Baja</b>
SHRS 2	Posible salida a internet de la placa embebida, sin embargo, existen otros métodos como módulos WiFi.		X	
SHRS 3	Capacidad de ejecutar lenguaje de programación de alto nivel para técnicas de visión artificial.	X		
SHRS 6	Debido a lo planteado en el presente proyecto se requiere una placa embebida.	X		
SHRS 10	Mismo requisito que SHRS2.	X		

SHRS 11	Memoria RAM entre 2Gb a 4gb para correr el algoritmo más rápidamente, se coloca una opción media debido a que un procesamiento de CPU o GPU tiene una mayor relevancia.	X
SHRS 12	Rendimiento de CPU igual o mayor a 13,5Gflop para tener un procesamiento aceptable para su fin, está en medio debido a que una tarjeta GPU puede tener una función similar.	X
SHRS 13	Ranura MicroSD para almacenamiento de base de datos de entrenamiento del sistema.	X
SiRS 15	Tamaño reducido para fácil manipulación.	X

### Cámara con acople a microscopio

En la siguiente Tabla 46 se puede observar el análisis de la importancia de los requisitos, ya definidos, que se toman en cuenta para la elección de una cámara digital para el sistema de conteo de esporas mediante técnicas de visión artificial.

Tabla 46

*Análisis requisitos para cámara con acople a microscopio*

Análisis		Alta	Med	Baja
Requisito				
SsRS 2	Se coloca una importancia media debido a que el sistema no cuenta figuras no concordantes con esporas de hongo trichoderma.		X	
SiRS 13	Salida mediante conector USB para comunicación con placa embebida.	X		
SiRS 16	Protección externa a manipulación y a una temperatura controlada. Se de una importancia media debido a que en su estación de trabajo no se mueve mucho y tiene una temperatura controlada.		X	
SHRS 2	Compatibilidad en sistema Linux es de gran importancia debido a que este se usa en el sistema al tener una política de código abierto.	X		
SHRS 7	Acople a microscopio de suma importancia debido a que las esporas de hongo trichoderma son microscópicas.	X		

## Protección externa

En la siguiente Tabla 47 se puede observar el análisis de la importancia de los requisitos, ya definidos, que se toman en cuenta para la protección externa para el sistema de conteo de esporas mediante técnicas de visión artificial.

Tabla 47

Análisis de requisitos de protección externa del sistema

	Análisis	Alta	Med	Baja
<b>Requisito</b>				
SiRS 15	Tamaño reducido con el fin de no complicar la manipulación del sistema por parte de los trabajadores.		X	
SiRS 16	Protección frente a manipulación de usuarios, importante debido a que varios usuarios calificados usarían este dispositivo.	X		
SiRS 17	Se colocó en una importancia media debido a que no es una temperatura peligrosa para el uso de placas embebidas.		X	

## Lenguaje de programación

En la siguiente Tabla 48 se puede observar el análisis de la importancia de los requisitos, ya definidos, que se toman en cuenta para la elección del lenguaje de alto nivel Python para el sistema de conteo de esporas mediante técnicas de visión artificial.

Tabla 48

Análisis requisitos para lenguaje de programación

	Análisis	Alta	Med	Baja
<b>Requisito</b>				
SHRS 15	Compatibilidad con librerías para correr algoritmos requeridos en el sistema.	X		
SHRS 16	Tener la posibilidad de usar licencias de libre uso, se considera de importancia media debido a que existe la posibilidad de comprar licencias.		X	
SHRS 18	Compatibilidad entre sistema operativos, software y hardware del sistema, se les brinda una importancia media debido a que cámaras echas para acoplarse con microscopio ciertas veces cierran su uso a aplicaciones fuera de las echas específicamente para estas.		X	
SHRS 19	Lenguaje de programación de alto nivel, para facilitar el correr el algoritmo de visión artificial.	X		

SHRS 22	IDE para facilitar la programación y tener un código más ordenado. Se coloca como importancia bajo debido a que se puede realizar esto en ambientes de programación diferentes.	X
---------	---	---

### Técnica de visión artificial

En la siguiente Tabla 49 se puede observar el análisis de la importancia de los requisitos, ya definidos, que se toman en cuenta para la elección de la técnica de visión artificial para el sistema de conteo de esporas mediante técnicas de visión artificial.

Tabla 49

*Análisis requisitos para lenguaje de programación*

Requisito	Análisis	Alta	Med	Baja
SHRS 23	Procesamiento de imagen, esencial para aplicar técnicas de visión artificial.	X		
SHRS 24	Parámetros compartidos considerado de importancia media debido a que permite un ahorro de memoria en disco (limitada en placas embebidas) al permitir compartir la base de datos de entrenamiento para diferentes algoritmos del sistema presente.		X	
SHRS 25	Relación espacial, se coloca una importancia mayor debido a que la relación entre el espacio y los eventos que ocurren en él, en este caso una imagen y varias detecciones de esporas.	X		

### Plataforma en nube compatible con Edge Computing

En la siguiente Tabla 50 se puede observar el análisis de la importancia de los requisitos, ya definidos, que se toman en cuenta para la elección de la plataforma de Edge Computing para el sistema de conteo de esporas mediante técnicas de visión artificial.

Tabla 50

Análisis requisitos para plataforma Edge Computing

<b>Requisito</b>	<b>Análisis</b>	<b>Alta</b>	<b>Med</b>	<b>Baja</b>
SHRS26	Código abierto, considerado de suma importancia debido a que se necesita total libertad de subir datos a la nube donde se encuentre su servidor.	X		
SHRS27	Compatibilidad con Python, se le considera esencial debido a que este es el lenguaje de programación que se utiliza en el sistema.	X		
SHRS28	Resultados subidos en servidor publicado en internet y de fácil acceso.	X		
SHRS29	Subida de datos mediante técnica HTTP, considerado de importancia media debido a que se puede utilizar otras tecnologías de IoT como MQTT.		X	

Firmas de autorización

<p>Ing. Fabián Geovanny Cuzme Rodríguez</p>  <p>_____</p> <p>Director</p>	<p>Sr. Esteban Marcelo Noboa Delgado</p>  <p>_____</p> <p>Estudiante</p>
--	--

## ANEXO E: Código del sistema

El ANEXO presente se divide en dos códigos, debido a que el proyecto tiene dos etapas principales, que son algoritmo de entrenamiento y código del sistema de detección y conteo. Cabe recordar que las acciones de entrenamiento son ejecutadas en la plataforma Google Colab y el código del sistema de detección y conteo corre en la placa Nvidia Jetson Nano.

### *iii. Código de entrenamiento*

En este código es necesario aclarar que se debe descargar anteriormente un archivo comprimido que contiene el sistema Dark net, este sirve para base para realizar la ejecución del modelo de aprendizaje o entrenamiento para el uso de Tiny Yolo:

```
!nvidia-smi
from google.colab import drive
drive.mount('/content/drive')
import os
os.chdir("/content/drive/My Drive/Trichoderma_v2/")
!chmod a+r /usr/local/cuda/include/cudnn.h
%cd darknet_v2/
!sed -i 's/OPENCV=0/OPENCV=1/' Makefile
!sed -i 's/GPU=0/GPU=1/' Makefile
!sed -i 's/CUDNN=0/CUDNN=1/' Makefile
!make
#!./darknet detector train custom_cfg/espora.data custom_cfg/espora.cfg custom_cfg/
darknet53.conv.74 -dont_show
!./darknet detector train custom_cfg/espora.data custom_cfg/espora.cfg backup/espora_
a_last.weights -dont_show
```

### *iv. Código del sistema de detección y conteo*

```
#Importamos librerías
import cv2 as cv
import numpy as np
import json
import pprint
import requests

from tkinter import *
from tkinter import filedialog
from PIL import Image
from PIL import ImageTk
import imutils

headers = {'Content-type': 'application/json'}
```

```

global image
global image1

# Creamos la ventana principal
root = Tk()

# Label donde se presentará la imagen de entrada
lblInputImage = Label(root)
lblInputImage.grid(column=0, row=2)

# Label donde se presentará la imagen de salida
lblOutputImage = Label(root)
lblOutputImage.grid(column=1, row=1, rowspan=6)

# Creamos los radios buttons y la ubicación que estos ocuparán
selected = IntVar()

rad1 = Radiobutton(root, text='Deteccion solución 1 a 100ml', width=25,value=1,
variable=selected, command= deteccion)

rad1.grid(column=0, row=4)

# Creamos el botón para elegir la imagen de entrada
btn = Button(root, text="Elegir imagen", width=25, command=elegir_imagen)
btn.grid(column=0, row=0, padx=5, pady=5)

root.mainloop()

#Funcion de visualizacion de imagen
def elegir_imagen():
    # Especificar los tipos de archivos, para elegir solo a las imágenes
    path_image = filedialog.askopenfilename(filetypes = [
        ("image", ".jpeg"),
        ("image", ".png"),
        ("image", ".jpg")])
    if len(path_image) > 0:
        global image
        global image1
        # Leer la imagen de entrada
        image = cv.imread(path_image)
        image1 = cv.imread(path_image)
        # Para visualizar la imagen de entrada en la GUI
        imageToShow= imutils.resize(image, width=120)
        imageToShow = cv.cvtColor(imageToShow, cv.COLOR_BGR2RGB)
        im = Image.fromarray(imageToShow )
        img1 = ImageTk.PhotoImage(image=im)

```

```

lblInputImage.configure(image=img1)
lblInputImage.image = img1
# Label IMAGEN DE ENTRADA
lblInfo1 = Label(root, text="IMAGEN DE ENTRADA:")
lblInfo1.grid(column=0, row=1, padx=5, pady=5)
# Al momento que leemos la imagen de entrada, vaciamos
# la imagen de salida y se limpia la selección de los
# radiobutton
lblOutputImage.image = ""
selected.set(0)

def subir(esporas):
    #Conexion con sistema IoT
    data = urllib.request.urlopen
    ("https://api.thingspeak.com/update?api_key=OJ6HVVCFAF1TDCQW&field1=0"+esporas);

#funcion para dibujar los objetos encontrados y conexion con IoT
def findObjects(outputs,img):
    hT, wT, cT = img.shape #alto, ancho y canales de img
    bbox = []
    classIds = []
    confs = []
    for output in outputs:
        for det in outputs[output]:
            scores = det[4:] #guardar en scores desde posicion 4 (%det,categoria)
            classId = np.argmax(scores) #sacar valores mas altos de scores
            confidence = scores[classId]
            if confidence > confThreshold: #(confidence>40%)
                w,h = int(det[2]*wT) , int(det[3]*hT) #medidas ancho y altura
                x,y = int((det[0]*wT)-w/2) , int((det[1]*hT)-h/2) #medidas
coordenada x, y
                bbox.append([x,y,w,h])
                classIds.append(classId) #guardado %Det > 40%
                confs.append(float(confidence)) #guardado %Det > 40%
            indices = cv.dnn.NMSBoxes(bbox, confs, confThreshold, nmsThreshold)
            #Eliminar exceso de rectángulos resaltantes con una objetividad < "0.2"
            (objetividad es operación intersección sobre unión de los cuadros)
            esporas=len(indices) #contador

```

```

for i in indices:
    i = i[0]
    box = bbox[i] #recorrido de bbox
    x, y, w, h = box[0], box[1], box[2], box[3]
    cv.rectangle(img, (x, y), (x+w,y+h), (150, 150 , 150), 2) #creacion
rectangulo
    cv.putText(img,f'{classNames[classIds[i]].upper()} {int(confs[i]*100)}%',
               (x, y), cv.FONT_HERSHEY_SIMPLEX, 0.45, (0, 255, 255), 1) #poner
nombre de clase y %Det en rectángulo
    cv.putText(img, 'Cantidad:'+str(esporas)+' esporas', (30,30),
cv.FONT_HERSHEY_SIMPLEX, 1, (0,255,0),2) #escribir contador en img

    subir(str(esporas)) #operacion subir contador a nube

whT = 832 # ancho y alto de la imagen
confThreshold =0.40 # umbral de precision o conincidencia
nmsThreshold= 0.2 # Umbral suprecision no maxima
#Cargamos las clases o etiquetas del detector
classesFile = "espora.names"
classNames = []
with open(classesFile, 'rt') as f:
    classNames = f.read().rstrip('\n').split('\n')
## Configuramos nuestra Red
modelConfiguration = "espora.cfg" #Archivo de arquitectura de la Red
modelWeights = "espora_30000.weights" # Pesos de la Red
net = cv.dnn.readNetFromDarknet(modelConfiguration, modelWeights) # Configuracion
de la red
net.setPreferableBackend(cv.dnn.DNN_BACKEND_OPENCV) # configuramos opencv por
detras
net.setPreferableTarget(cv.dnn.DNN_TARGET_CPU) # configuramos que trabaje con
cpu

def deteccion():
    global image
    if selected.get() == 1:
        img=image #Imagen
        blob = cv.dnn.blobFromImage(img, 1/255, (whT, whT), [0, 0, 0], 1,
crop=False) #Tratamiento imagen
        net.setInput(blob) #enviar imagen tratada a red Tiny Yolo
        layersNames = net.getLayerNames() #Extraer operaciones de entrenamiento de
red Yolo

```

```

        #print(net.getUnconnectedOutLayers())

        outputNames = [(layersNames[i[0]-1]) for i in
net.getUnconnectedOutLayers()] #Recorrido de operaciones #2 tipos de capas de
salida (yolo_16 y yolo_23) relacionadas con dos escalas (26x26 y 13x13).

        #print(outputNames) #reenvio de operaciones de entrenamiento

        outputs = net.forward(outputNames) #obtencion matriz de deteccion [(2028,6)
y (8112,6)].

        #2028 y 8112 representa todas las predicciones y 6 es número de elementos
matriz: [x, y, w, h, %Confianza, categoría].

        findObjects(outputs,img)

# Para visualizar la imagen en lblOutputImage en la GUI
img = cv.cvtColor(img, cv.COLOR_BGR2RGB)
img= imutils.resize(img, width=832)
im = Image.fromarray(img)
img1 = ImageTk.PhotoImage(image=im)
lblOutputImage.configure(image=img1)
lblOutputImage.image = img1

        # Label IMAGEN DE SALIDA
lblInfo3 = Label(root, text="IMAGEN DE SALIDA:", font="bold")
lblInfo3.grid(column=1, row=0, padx=5, pady=5)

# Para visualizar la imagen en lblOutputImage en la GUI
img2 = cv.cvtColor(img1, cv.COLOR_BGR2RGB)
img2= imutils.resize(img2, width=875)
im = Image.fromarray(img2)
img1 = ImageTk.PhotoImage(image=im)
lblOutputImage.configure(image=img1)
lblOutputImage.image = img1

        # Label IMAGEN DE SALIDA
lblInfo3 = Label(root, text="IMAGEN DE SALIDA:", font="bold")
lblInfo3.grid(column=1, row=0, padx=5, pady=5)

#SUBIR resultado a nube
subir(str(esporas))

```

## ANEXO F: Lista de asistencia a capacitación



### LISTA ASISTENCIA DE INDUCCIÓN

**Evento:** Inducción para el uso del sistema de detección y conteo de esporas de hongo *Trichoderma*

**Lugar del evento:** Laboratorio de BiosebOrganics

**Fecha:** Miércoles 15 de junio del 2022

No	Nombres y apellidos	Cargo	Firma	
			Entrada	Salida
1	Mercy Jeanelh Escobar Reyes	laboratorista		
2	Luis Gabriel Cunipalla Wina	Jefe de planta		
3	Manuel Misael Bravo Trilema	laboratorista		
4	Alexandro Ortiz Villacis	laboratorista		

## REFERENCIAS

- Ahmed, O., & Brifcani, A. (2019). Gene Expression Classification Based on Deep Learning. *4th Scientific International Conference Najaf, SICN 2019*, 145–149. <https://doi.org/10.1109/SICN47020.2019.9019357>
- AIA. (1984). *Automated Imaging Association*. <https://www.automate.org/vision>
- Alexopoulos, C. J., Mims, C. W., & Blackwell, M. (1996). *Introductory mycology. Introductory Mycology., Ed.4.*
- Alfáu Ascuasiati, A. (2012). *Plagas Domésticas: Vol. Primera edición*. Oasis Colonial. [https://books.google.com.ec/books?hl=es&lr=&id=29jnczPBAqgC&oi=fnd&pg=PA4&dq=control+de+plagas+en+ambito+domestico&ots=FzC5InKBcV&sig=aBSSFsohAmsRucd3IIrE7di8qg4&redir\\_esc=y#v=onepage&q=control de plagas en ambito domestico&f=false](https://books.google.com.ec/books?hl=es&lr=&id=29jnczPBAqgC&oi=fnd&pg=PA4&dq=control+de+plagas+en+ambito+domestico&ots=FzC5InKBcV&sig=aBSSFsohAmsRucd3IIrE7di8qg4&redir_esc=y#v=onepage&q=control+de+plagas+en+ambito+domestico&f=false)
- Algar Díaz, J., & Fernández de Sevilla Vellón, M. (2019). *Introducción practica a la programación con Python*. Editorial Universidad de Alcalá. <https://elibro.net/es/ereader/uladech/124259?page=20>
- Alvarez-Ramos, C. M., Nino, E., & Santos, M. (2013). Automatic classification of *Nosema* pathogenic agents through machine vision techniques and kernel-based vector machines. *2013 8th Computing Colombian Conference, 8CCC 2013*. <https://doi.org/10.1109/ColombianCC.2013.6637516>
- Alvear-Puertas, V., Rosero-Montalvo, P., Peluffo-Ordóñez, D., & Pijal-Rojas, J. (2017). Internet de las Cosas y Visión Artificial, Funcionamiento y Aplicaciones: Revisión de Literatura. *Enfoque UTE*, 8(1), 244–256. <https://doi.org/10.29019/enfoqueute.v8n1.121>
- Arévalo, P. A. (2018). *Determinación de la capacidad de degradación de cianuro proveniente de la industria minera del cantón cuenca con Trichoderma spp.* 67.
- Argumedo Delira, R., Alejandro, A., Ferrera Cerrato, R., & Pena Cabriales, J. (2009). El género fúngico *Trichoderma* y su relación con los contaminantes orgánicos e inorgánicos. *Revista Internacional de Contaminación Ambiental*. [http://www.scielo.org.mx/scielo.php?pid=S0188-49992009000400006&script=sci\\_arttext](http://www.scielo.org.mx/scielo.php?pid=S0188-49992009000400006&script=sci_arttext)
- Bissett, J. (1991). A revision of the genus *Trichoderma* . III. Section *Pachybasium* . *Canadian*

*Journal of Botany*, 69(11), 2373–2417. <https://doi.org/10.1139/b91-298>

- Bradski, G., & Kaehler, A. (2008). *Computer Vision con la biblioteca OpenCV*. O'Reilly Media, Inc. [https://books.google.com.ec/books?hl=es&lr=&id=seAgiOfu2EIC&oi=fnd&pg=PR3&dq=opencv&ots=hVG37mgJTc&sig=zO2T37Asolx2jYHLzxEceABS5Ho&redir\\_esc=y#v=onepage&q=opencv&f=false](https://books.google.com.ec/books?hl=es&lr=&id=seAgiOfu2EIC&oi=fnd&pg=PR3&dq=opencv&ots=hVG37mgJTc&sig=zO2T37Asolx2jYHLzxEceABS5Ho&redir_esc=y#v=onepage&q=opencv&f=false)
- Buleje, C. Y. C., Atencio, Y. T. P., & Tinoco, E. E. C. (2021). System with Optical Mark Recognition Based on Artificial Vision for the Processing of Multiple Selection Tests in School Competitions. *Latin America Computing Conference*, 172–177. <https://doi.org/10.1109/CLEI52000.2020.00027>
- Bustillo, A. E. (2004). *Prevención de los riesgos para la salud derivados del uso de plaguicidas en la agricultura* Compilado por International Centre for Pesticide Safety Diseño de portada Tuula Solasaari-Pekki, Instituto Finandés de Salud Ocupacional Corrección de pruebas Ineke. I (disminuir la dependencia de los plaguicidas en la agricultura).
- Caiza Sango, S. E. (2017). *EVALUACIÓN IN VITRO DE LA CAPACIDAD ANTAGÓNICA DE Trichoderma COMERCIAL (Trichoderma harzianum) Y Trichoderma NATIVO (Trichoderma sp.) FRENTE A LOS PATÓGENOS Alternaria sp., Fusarium oxysporum y Heterosporium echinolatum DEL CULTIVO DE CLAVEL (Dianthus ca.* Escuela Politécnica Nacional.
- Calderón Reina, M. J. (2017). *Diferentes estrategias para minimizar la contaminación por plaguicidas de aguas superficiales y subterráneas en zonas olivareras.* <https://dialnet.unirioja.es/servlet/tesis?codigo=146749&info=resumen&idioma=SPA>
- Castillo, J. (2015). *Trichoderma viride Pers. 1794.* In */BANCO DE IMÁGENES DE HONGOS DE LA PENÍNSULA IBÉRICA.* <https://www.biodiversidadvirtual.org/hongos/Trichoderma-viride-Pers.-1794-img102836.html>
- Chen, M., Challita, U., Saad, W., Yin, C., & Debbah, M. (2019). Artificial Neural Networks-Based Machine Learning for Wireless Networks: A Tutorial. *IEEE Communications Surveys and Tutorials*, 21(4), 3039–3071. <https://doi.org/10.1109/COMST.2019.2926625>
- Consejo Nacional de Planificación. (2017). Plan Nacional del Buen vivir 2017-2021.

- Educational Research*, 1, 150. <http://www.planificacion.gob.ec/wp-content/uploads/downloads/2017/07/Plan-Nacional-para-el-Buen-Vivir-2017-2021.pdf>
- Elorza, M. I. (2016). Control natural de plagas en huerto orgánico. *Ilustre Municipio de Santiago*. [http://www.munistgo.info/medioambiente/wp-content/uploads/2016/10/Control\\_de\\_plagas.pdf](http://www.munistgo.info/medioambiente/wp-content/uploads/2016/10/Control_de_plagas.pdf)
- Estrada, M. A. J., Jiménez, J. J. C., García, V. M. P., & Isaías, G. T. (2002). Evaluación del conteo total de ooquistes de *Eimeria tenella* a partir del ciego o heces, efectuado con la cámara de McMaster y el hemocitómetro de Neubauer. *Veterinaria México*, 33(1), 73–79. <https://www.redalyc.org/articulo.oa?id=42333108>
- FAR-EDGE. (2017). *Project: FAR-EDGE H2020*. <http://www.faredge.eu/#/>
- Farinango, H., Rosero-Montalvo, P., Peluffo-Ordóñez, D., & Cuzme-Rodríguez, F. (2016). Internet de las Cosas y Visión Artificial, Funcionamiento y Aplicaciones: Revisión de Literatura. *Enfoque UTE*, 8(1), 244–256. <https://doi.org/10.29019/enfoqueute.v8n1.121>
- Fumanal, S. (2013). *Distribución cosmopolita | Biodiversidad Virtual*. <https://www.biodiversidadvirtual.org/taxofoto/glosario/1679>
- García Laureano, R. (2020). *Identificación de los productos y medios empleados para el control de plagas* (E. T. FORMACIÓN (ed.); Diciembre). <https://elibro.net/es/ereader/utnorte/171603?page=7>
- Garrido, Á. (2020). *Los avances de la inteligencia artificial*. DYKINSON, S L.
- Gedeon, J., Brandherm, F., Egert, R., Grube, T., & Mühlhäuser, M. (2019). What the Fog? Edge Computing Revisited: Promises, Applications and Future Challenges. In *IEEE Access* (Vol. 7). <https://doi.org/10.1109/ACCESS.2019.2948399>
- Gil, M. (2019). *Azul de lactofenol: características, composición, preparación, usos*. 30 de Octubre. <https://www.lifeder.com/azul-de-lactofenol/>
- Gómez, T. (2017). Evaluación De Diferentes Medios De Cultivo Y Condiciones Para La Producción De Conidios De *Trichoderma Spp* Mediante Fermentación En Líquido Y Sólido. *Instituto Del Café de Costa Rica*, 15. [https://repositoriotec.tec.ac.cr/bitstream/handle/2238/7079/evaluacion\\_diferentes\\_medios\\_condiciones\\_produccion.pdf?sequence=1&isAllowed=y](https://repositoriotec.tec.ac.cr/bitstream/handle/2238/7079/evaluacion_diferentes_medios_condiciones_produccion.pdf?sequence=1&isAllowed=y)

- González, J. L. N. (2018). *Detección de incendios mediante identificación de humo con visión artificial en condiciones de iluminación variable*. <http://hdl.handle.net/20.500.11799/95189>
- Gutierrez, M., & Alayo, E. (2017). Degradación del insecticida alfacipermetrina por *Trichoderma harzianum*. *Véritas*, *16*(2015), 61–64.
- Hamet, P., & Tremblay, J. (2017). Artificial intelligence in medicine. *Metabolism: Clinical and Experimental*, *69*, S36–S40. <https://doi.org/10.1016/j.metabol.2017.01.011>
- Herrera Beltran, G. M. (2021). *Fermentación sólida en la industria alimentaria*. Trabajo. UNIVERSIDAD TÉCNICA DE AMBATO.
- Hispaviacion. (2020). *Los drones de búsqueda y rescate en Canadá utilizarán Inteligencia Artificial para optimizar las operaciones*. <https://www.hispaviacion.es/los-drones-de-busqueda-y-rescate-en-canada-utilizaran-inteligencia-artificial-para-optimizar-las-operaciones/>
- Infante, D., Martínez, B., Gonzáles, N., & Reyes, Y. (2019). MECANISMOS DE ACCIÓN DE *Trichoderma* FRENTE A HONGOS FITOPATÓGENOS. *Revista de Preservación Vegetal*, *24*, 14–21.
- IOTHOOK. (n.d.). *Open Source IOT Dashboard*. Retrieved January 21, 2022, from <https://iothook.com/en/>
- ISO/IEC/IEEE. (2011). ISO/IEC/IEEE 42010. *Functional Analysis and Its Applications*, *16*(4), 319–321. <https://doi.org/10.1007/BF01077867>
- Ivorra Martínez, E. (2021). *Introducción al software de visión artificial Sherlock*. <https://riunet.upv.es/handle/10251/165939>
- Jacas, J., Caballero, P., & Avilla, J. (2014). *El control biológico de plagas y enfermedades : la sostenibilidad de la agricultura mediterránea*. Universitat Jaume I. Servei de Comunicació i Publicacions. <https://elibro.net/es/ereader/utnorte/53255?page=122>.
- Lee, V. H., Hew, J. J., Leong, L. Y., Tan, G. W. H., & Ooi, K. B. (2020). Wearable payment: A deep learning-based dual-stage SEM-ANN analysis. *Expert Systems with Applications*, *157*, 113477. <https://doi.org/10.1016/J.ESWA.2020.113477>
- López de Mántaras Badia, R., & Meseguer González, P. (2017). *Inteligencia artificial*.

Editorial CSIC Consejo Superior de Investigaciones Científicas.

Madhavan, S., & Jones, T. (2017, September 8). Deep learning architectures . *IBM Developer*.

Malik, M. I. (2018). Google Colaboratory como alternativa para el procesamiento de una red neuronal convolucional. *International Journal of Advanced Research in Computer Science*, 9(2), 379–384. <https://doi.org/10.26483/IJARCS.V9I2.5760>

Maliza, R., & Elena, R. (2017). “Sistema De Reconocimiento Facial Con Visión Artificial Para Apoyar Al Ecu-911 Con La Identificación De Personas En La Lista De Los Más Buscados.” *Repo.Uta.Edu.Ec*, 130. <http://repo.uta.edu.ec/bitstream/handle/123456789/5301/Mg.DCEv.Ed.1859.pdf?sequence=3>

Marroquin, P. E. F., Leon, E. F. P., & Figueroa, F. A. V. (2004). *Sistema De Inspeccion Basado En Vision Artificial Aplicado a Un Proceso Industrial*.

Martinez, J. (2020). *Gradiente Descendiente para aprendizaje automático*. IArtificial.Net. <https://www.iartificial.net/gradiente-descendiente-para-aprendizaje-automatico/>

Maya Álvarez, M. A. (2017). *Caracterización y control de plagas en áreas edificadas y ajardinadas*. SEAG0110. IC Editorial.

Militante, S. V., Gerardo, B. D., & Dionisio, N. V. (2019). Plant Leaf Detection and Disease Recognition using Deep Learning. *2019 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE)*, 579–582. <https://doi.org/10.1109 / ECICE47484.2019.8942686>

Ministerio del Trabajo del Ecuador. (2021). Salarios Mínimos Sectoriales 2021. In [https://www.trabajo.gob.ec/wp-content/uploads/2020/12/ANEXO-1 “Estructuras-ocupacionales---salarios-m%C3%ADnimos-sectoriales-y-tarifas-sa.pdf?x42051](https://www.trabajo.gob.ec/wp-content/uploads/2020/12/ANEXO-1-Estructuras-ocupacionales---salarios-m%C3%ADnimos-sectoriales-y-tarifas-sa.pdf?x42051) (Issue 1).

Molnar, C. (2020). *Interpretable machine learning*. [https://books.google.com.ec/books?hl=es&lr&id=jBm3DwAAQBAJ&oi=fnd&pg=PP1&dq=machine+learning&ots=EguSXRCHYZ&sig=MivEaFSqN2XGSjJONR7ylLgXKNI&redir\\_esc=y&pli=1#v=onepage&q&f=false](https://books.google.com.ec/books?hl=es&lr&id=jBm3DwAAQBAJ&oi=fnd&pg=PP1&dq=machine+learning&ots=EguSXRCHYZ&sig=MivEaFSqN2XGSjJONR7ylLgXKNI&redir_esc=y&pli=1#v=onepage&q&f=false)

Montero, B., Cevallos, H., & Dávila, J. (2018). Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software. *Espirales Revista Multidisciplinaria de*

*Investigación* ISSN: 2550-6862, 2(17), 114–121.  
[https://www.researchgate.net/publication/327537074\\_Metodologias\\_agiles\\_frente\\_a\\_las\\_tradicionales\\_en\\_el\\_proceso\\_de\\_desarrollo\\_de\\_software](https://www.researchgate.net/publication/327537074_Metodologias_agiles_frente_a_las_tradicionales_en_el_proceso_de_desarrollo_de_software)

- Muñoz Cuaical, S. D. (2016). Evaluación de la eficacia de *Trichoderma sp* y *Pseudomona sp* para biorremediación de suelos contaminados con hidrocarburos. *Universidad Politécnica Salesiana Sede-Quito*, 125.
- Naranjo Silva, J. L. (2016). *DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA PARA LA CONTABILIZACIÓN AUTOMÁTICA DE UNIDADES FORMADORAS DE COLONIAS BACTERIANAS: COLIFORMES, MESÓFILOS AEROBIOS, BACILOS, ESTAFILOCOS Y LEVADURAS USANDO TÉCNICAS DE VISIÓN ARTIFICIAL* [Universidad de las Fuerzas Armadas (ESPE)]. <http://repositorio.espe.edu.ec/bitstream/21000/12017/1/T-ESPE-053357.pdf>
- Nedumaan, J., Binford, T., Lepika, J., Tisa, J., Ruby, J., & Kumar, J. (2019). *Modern Deep Learning and Advanced Computer Vision: A Perspective Approach* (I. Published (ed.)). [https://books.google.com.ec/books/about/Modern\\_Deep\\_Learning\\_and\\_Advanced\\_Comput.html?id=70q8zAEACAAJ&redir\\_esc=y](https://books.google.com.ec/books/about/Modern_Deep_Learning_and_Advanced_Comput.html?id=70q8zAEACAAJ&redir_esc=y)
- Núñez Sánchez, F. J. (2016). *Diseño de un sistema de reconocimiento automático de matrículas de vehículos mediante una red neuronal convolucional*. <http://openaccess.uoc.edu/webapps/o2/handle/10609/52222>
- Oliveira, R. (2019). Hemograma - Como fazer e interpretar. In Amolca (Ed.), *LMP Editora* (Segunda).
- OMS, O. M. de la salud, & FAO, O. ación de las N. U. para la A. y A. (2010). *Código internacional de conducta para la distribución y utilización de plaguicidas*.
- Ordoñez, J. F., & Oñate, L. (2016). Detector De Bordes De Imágenes Usando Un Microcontrolador Arm. *Ingenius*, 16, 30. <https://doi.org/10.17163/ings.n16.2016.04>
- Oropeza Clavel, C. A. (2007). *Modelado y Simulacion de un Sistema de Deteccion de Intrusos Utilizando Redes Neuronales Recurrentes*.
- Ortega Candell, J. M. (2018). *Hacking ético con herramientas Python*. RA-MA Editorial. <https://elibro.net/es/ereader/utnorte/106513?page=16>
- Pamoarato, M., Begonja, S., & Espósito, M. (2018). *Introducción a la química Hidrocarburos*,

*alimentos y procesos industriales.* Primera Edición.  
<https://elibro.net/es/ereader/utnorte/138495>

- Pasquel, C. P. P. (2018). *Diseño De Un Sistema De Reconocimiento Automático De Vehículos Mediante El Uso De Redes Neuronales Profundas (Dnn)*. Universidad Técnica del Norte.
- Pastor, E. (2019). SomData, visión artificial para detectar los microorganismos del agua - Valencia Plaza. *Valencia Plaza*. <https://valenciaplaza.com/som-data-escaner-deteccion-microorganismos-agua>
- Patricia, J., Gamboa, Z., Alexandra, C., & Arreaga, L. (2018). Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software. Evolution of the Methodologies and Models used in Software Development. *INNOVA Research Journal*, 3(10), 20–33.
- Peña, A. R., & Sanchez, J. J. C. (2020). *Prototipo de geolocalización y reconocimiento aplicado a especies*. 89.
- Pinto Rodríguez, J. (2018). *Identificación de los productos y medios empleados para el control de plagas. SEAG0110*. 179.
- Polit, U., Val, C. D. E., & Catal, L. (2020). *Diseño de un proceso de producción de ácido acrílico , vía acroleína*. 1–71.
- Portero Donoso, P., & Mena Mena, B. (2017). “DESARROLLO DE UN PROTOTIPO PARA EL CONTROL DE CALIDAD DE LA CARNE BOVINA DETERMINADA POR SUS CARACTERÍSTICAS ORGANOLÉPTICAS, BASADO EN UN SISTEMA AUTOMÁTICO DE INSPECCIÓN POR VISIÓN ARTIFICIAL. *ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO*.
- Project Management Institute. (2017). *A guide to the project management body of knowledge PMBOK GUIDE* (Sexta). Project Management Institute, Inc.
- RAE. (2020). *Diccionario de la Real Academia de la Lengua Española*. Fundación La Caixa. <https://dle.rae.es/procesador?m=form>
- Rampérez Barrajón, J. (2021). El ácido poliláctico (PLA): Un material innovador en la actualidad. *Universidad Almeriense*.
- Raschka, S., Patterson, J., & Nolet, C. (2020). *information Machine Learning in Python: Main Developments and Technology Trends in Data Science, Machine Learning, and Artificial*

*Intelligence*. <https://doi.org/10.3390/info11040193>

Reyes, N. (2017). *DISEÑO ÓPTIMO DE PLANTAS DE FERMENTACIÓN DISCONTINUA* [UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA DEPARTAMENTO]. <http://tinyurl.com/glmrldv>

Rodríguez, F. (2011). *Contacto – Bioseb Organics*. <https://www.bioseb.org/contacto>

Roman Bueno, J., & Gonzalez Mantilla, K. (2017). *Sistemas embebidos y Hardware libre*. <http://wiki.sc3.uis.edu.co/images/e/e1/GR7.pdf>

Sánchez Hernández, L., Rosa, A. M., Rosique Gil, J., & Pachero Figueroa, C. (2018). Diversidad del género *Trichoderma* (Hypocraceae) en un Área Natural Protegida en Tabasco, México. *Acta Botanico Mexicana* 123, 167–182.

Sánchez Salinas, E., & Ortiz Hernández, L. (2018). Riesgos y estrategias en el uso de plaguicidas. *Inventio, La Génesis de La Cultura Universitaria En Morelos*, 11(14), 21–27.

Sandoval, M. C., & Belesansky, C. (2020). *Producción artesanal del hongo antagónico Trichoderma Persoon en sustrato sólido*. 7(3), 55–64.

Sanipatín, P. (2021). *RECONOCIMIENTO DE EXPRESIONES FACIALES A TRAVÉS DE UN ANÁLISIS DE PATRONES DE MOVIMIENTOS MUSCULARES FACIALES POR MEDIO DE TÉCNICAS DE APRENDIZAJE PROFUNDO*. Universidad Técnica del Norte.

Shi, W., & Dustdar, S. (2016). *The Promise of Edge Computing*. 18, Cloud Cover.

Silva, N., & Alvarez, S. (2013). Producción Artesanal y Control de Calidad del hongo antagonista *Trichoderma*. In *PRODUCCION ARTESANAL DE TRICHODERMA* (p. 48). [https://www.academia.edu/11343364/Producción\\_Artesanal\\_y\\_Control\\_de\\_Calidad\\_del\\_hongo\\_antagonista\\_Trichoderma](https://www.academia.edu/11343364/Producción_Artesanal_y_Control_de_Calidad_del_hongo_antagonista_Trichoderma)

Sittón-Candanedo, I., Alonso, R. S., Muñoz, L., & Rodríguez-González, S. (2019). Arquitecturas de Referencia Edge Computing para la Industria 4.0: una revisión. *Arquitecturas de Referencia Edge Computing Para La Industria 4.0: Una Revisión. Edge Computing Reference Architectures for Industry 4.0: A Review, III Congreso Internacional en Inteligencia Ambiental, Ingeniería de Software y Salud Electrónica y Móvi*, 16–23. <https://ridda2.utp.ac.pa/bitstream/handle/123456789/6471/Proceeding->

AmITIC-2019-Pereira-Colombia.pdf?sequence=1&isAllowed=y

Tecnologiavao. (2019). *El papel de la visión artificial en la industria automotriz*.  
<https://tecnologiaparalaindustria.com/el-papel-de-la-vision-artificial-en-la-industria-automotriz/>

Terou, H., & Parr, J. (1967). MICROORGANISMOS BENÉFICOS Y EFECTIVOS PARA UNA AGRICULTURA Y MEDIO AMBIENTE SOSTENIBLE. *Angewandte Chemie International Edition*, 6(11), 951–952. <https://itscv.edu.ec/wp-content/uploads/2018/10/MICROORGANISMOS-DEL-SUELO-PARA-LA-AGRICULTURA.pdf>

The MathWorks, I. (2022). *Thing Speak*. <https://thingspeak.com/>

Thew, R., Calarco, T., Roorda, P. D., Ca, O., Jones, K. P., Ca, K., Friesen, P., Ca, O., Patent, U. S., Ci, U. S., ITU, Systems, D., ITU-T, 129/Menkes/SK/II/2008, M. K. R. I. N. :, מזלי, ג., Roberts, A., Mujtaba, A., Mujtaba, A., Mujtaba, A., ... Seifert, F. (2015). Estudio del adecuado crecimiento del hongo trichoderma harzianum y trichoderma hamatum en sustrato sólido [UNIVERSIDAD CENTRAL DEL ECUADOR]. In *Metrologia* (Vol. 53, Issue 5).  
[http://publicacoes.cardiol.br/portal/ijcs/portugues/2018/v3103/pdf/3103009.pdf%0Ahttp://www.scielo.org.co/scielo.php?script=sci\\_arttext&pid=S0121-75772018000200067&lng=en&tlng=en&SID=5BQIj3a2MLaWUV4OizE%0Ahttp://scielo.iec.pa.gov.br/scielo.php?script=sci\\_](http://publicacoes.cardiol.br/portal/ijcs/portugues/2018/v3103/pdf/3103009.pdf%0Ahttp://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0121-75772018000200067&lng=en&tlng=en&SID=5BQIj3a2MLaWUV4OizE%0Ahttp://scielo.iec.pa.gov.br/scielo.php?script=sci_)

Vázquez Nin, G. (2000). *Introducción a la Microscopía Electrónica Aplicada a Las Ciencias Biológicas*. Universidad México Americana del Norte.  
<https://books.google.es/books?hl=es&lr=&id=JNoMEWVRBmkC&oi=fnd&pg=PA7&dq=+MICROSCOPIA+&ots=CooOnaK0SD&sig=U0tn3vpeEjuWmdZJITZQLky8Zzo#v=onepage&q=MICROSCOPIA&f=false>

Vencoel. (2020, July 2). *Inteligencia aplicada a Sistemas de Visión Artificial: Evolución y Hardware - Venco Electronica Industrial*. <https://www.vencoel.com/inteligencia-aplicada-a-sistemas-de-vision-artificial-evolucion-y-hardware/>

Woo, S. L., Ruocco, M., Vinale, F., Nigro, M., Marra, R., Lombardi, N., Pascale, A., Lanzuise, S., Manganiello, G., & Lorito, M. (2014). Trichoderma-based Products and their Widespread Use in Agriculture. *The Open Mycology Journal*, 8(1), 71–126.

<https://doi.org/10.2174/1874437001408010071>

Yordanov, Y. E., & Mladenov, V. M. (2018). Humanoid Robot Detecting Animals via Neural Network. *2018 14th Symposium on Neural Networks and Applications, NEUREL 2018*. <https://doi.org/10.1109/NEUREL.2018.8587017>

Ahmed, O., & Brifcani, A. (2019). Gene Expression Classification Based on Deep Learning. *4th Scientific International Conference Najaf, SICN 2019*, 145–149. <https://doi.org/10.1109/SICN47020.2019.9019357>

AIA. (1984). *Automated Imaging Association*. <https://www.automate.org/vision>

Alexopoulos, C. J., Mims, C. W., & Blackwell, M. (1996). *Introductory mycology. Introductory Mycology., Ed.4.*

Alfáu Ascuasiati, A. (2012). *Plagas Domésticas: Vol. Primera edición*. Oasis Colonial. [https://books.google.com.ec/books?hl=es&lr=&id=29jnczPBAqgC&oi=fnd&pg=PA4&dq=control+de+plagas+en+ambito+domestico&ots=FzC5InKBcV&sig=aBSSFsohAmsRucd3IIrE7di8qg4&redir\\_esc=y#v=onepage&q=control de plagas en ambito domestico&f=false](https://books.google.com.ec/books?hl=es&lr=&id=29jnczPBAqgC&oi=fnd&pg=PA4&dq=control+de+plagas+en+ambito+domestico&ots=FzC5InKBcV&sig=aBSSFsohAmsRucd3IIrE7di8qg4&redir_esc=y#v=onepage&q=control%20de%20plagas%20en%20ambito%20domestico&f=false)

Algar Díaz, J., & Fernández de Sevilla Vellón, M. (2019). *Introducción practica a la programación con Python*. Editorial Universidad de Alcalá. <https://elibro.net/es/ereader/uladech/124259?page=20>

Alvarez-Ramos, C. M., Nino, E., & Santos, M. (2013). Automatic classification of *Nosema* pathogenic agents through machine vision techniques and kernel-based vector machines. *2013 8th Computing Colombian Conference, 8CCC 2013*. <https://doi.org/10.1109/ColombianCC.2013.6637516>

Alvear-Puertas, V., Rosero-Montalvo, P., Peluffo-Ordóñez, D., & Pijal-Rojas, J. (2017). Internet de las Cosas y Visión Artificial, Funcionamiento y Aplicaciones: Revisión de Literatura. *Enfoque UTE*, 8(1), 244–256. <https://doi.org/10.29019/enfoqueute.v8n1.121>

Arévalo, P. A. (2018). *Determinación de la capacidad de degradación de cianuro proveniente de la industria minera del cantón cuenca con Trichoderma spp.* 67.

Argumedo Delira, R., Alejandro, A., Ferrera Cerrato, R., & Pena Cabriales, J. (2009). El género fúngico *Trichoderma* y su relación con los contaminantes orgánicos e inorgánicos. *Revista Internacional de Contaminación Ambiental*.

[http://www.scielo.org.mx/scielo.php?pid=S0188-49992009000400006&script=sci\\_arttext](http://www.scielo.org.mx/scielo.php?pid=S0188-49992009000400006&script=sci_arttext)

- Bissett, J. (1991). A revision of the genus *Trichoderma* . III. Section *Pachybasium* . *Canadian Journal of Botany*, 69(11), 2373–2417. <https://doi.org/10.1139/b91-298>
- Bradski, G., & Kaehler, A. (2008). *Computer Vision con la biblioteca OpenCV*. O'Reilly Media, Inc. [https://books.google.com.ec/books?hl=es&lr=&id=seAgiOfu2EIC&oi=fnd&pg=PR3&dq=opencv&ots=hVG37mgJTc&sig=zO2T37Asolx2jYHLzxEceABS5Ho&redir\\_esc=y#v=onepage&q=opencv&f=false](https://books.google.com.ec/books?hl=es&lr=&id=seAgiOfu2EIC&oi=fnd&pg=PR3&dq=opencv&ots=hVG37mgJTc&sig=zO2T37Asolx2jYHLzxEceABS5Ho&redir_esc=y#v=onepage&q=opencv&f=false)
- Buleje, C. Y. C., Atencio, Y. T. P., & Tinoco, E. E. C. (2021). System with Optical Mark Recognition Based on Artificial Vision for the Processing of Multiple Selection Tests in School Competitions. *Latin America Computing Conference*, 172–177. <https://doi.org/10.1109/CLEI52000.2020.00027>
- Bustillo, A. E. (2004). *Prevención de los riesgos para la salud derivados del uso de plaguicidas en la agricultura* Compilado por International Centre for Pesticide Safety Diseño de portada Tuula Solasaari-Pekki, Instituto Finlandés de Salud Ocupacional Corrección de pruebas Ineke. I (disminuir la dependencia de los plaguicidas en la agricultura).
- Caiza Sango, S. E. (2017). *EVALUACIÓN IN VITRO DE LA CAPACIDAD ANTAGÓNICA DE Trichoderma COMERCIAL (Trichoderma harzianum) Y Trichoderma NATIVO (Trichoderma sp.) FRENTE A LOS PATÓGENOS Alternaria sp., Fusarium oxysporum y Heterosporium echinolatum DEL CULTIVO DE CLAVEL (Dianthus ca.* Escuela Politécnica Nacional.
- Calderón Reina, M. J. (2017). *Diferentes estrategias para minimizar la contaminación por plaguicidas de aguas superficiales y subterráneas en zonas olivareras.* <https://dialnet.unirioja.es/servlet/tesis?codigo=146749&info=resumen&idioma=SPA>
- Castillo, J. (2015). *Trichoderma viride Pers. 1794. In /BANCO DE IMÁGENES DE HONGOS DE LA PENÍNSULA IBÉRICA.* <https://www.biodiversidadvirtual.org/hongos/Trichoderma-viride-Pers.-1794-img102836.html>
- Chen, M., Challita, U., Saad, W., Yin, C., & Debbah, M. (2019). Artificial Neural Networks-

- Based Machine Learning for Wireless Networks: A Tutorial. *IEEE Communications Surveys and Tutorials*, 21(4), 3039–3071. <https://doi.org/10.1109/COMST.2019.2926625>
- Consejo Nacional de Planificación. (2017). Plan Nacional del Buen vivir 2017-2021. *Educational Research*, 1, 150. <http://www.planificacion.gob.ec/wp-content/uploads/downloads/2017/07/Plan-Nacional-para-el-Buen-Vivir-2017-2021.pdf>
- Elorza, M. I. (2016). Control natural de plagas en huerto orgánico. *Ilustre Municipio de Santiago*. [http://www.munistgo.info/medioambiente/wp-content/uploads/2016/10/Control\\_de\\_plagas.pdf](http://www.munistgo.info/medioambiente/wp-content/uploads/2016/10/Control_de_plagas.pdf)
- Estrada, M. A. J., Jiménez, J. J. C., García, V. M. P., & Isaías, G. T. (2002). Evaluación del conteo total de ooquistes de *Eimeria tenella* a partir del ciego o heces, efectuado con la cámara de McMaster y el hemocitómetro de Neubauer. *Veterinaria México*, 33(1), 73–79. <https://www.redalyc.org/articulo.oa?id=42333108>
- FAR-EDGE. (2017). *Project: FAR-EDGE H2020*. <http://www.faredge.eu/#/>
- Farinango, H., Rosero-Montalvo, P., Peluffo-Ordóñez, D., & Cuzme-Rodríguez, F. (2016). Internet de las Cosas y Visión Artificial, Funcionamiento y Aplicaciones: Revisión de Literatura. *Enfoque UTE*, 8(1), 244–256. <https://doi.org/10.29019/enfoqueute.v8n1.121>
- Fumanal, S. (2013). *Distribución cosmopolita | Biodiversidad Virtual*. <https://www.biodiversidadvirtual.org/taxofoto/glosario/1679>
- García Laureano, R. (2020). *Identificación de los productos y medios empleados para el control de plagas* (E. T. FORMACIÓN (ed.); Diciembre). <https://elibro.net/es/ereader/utnorte/171603?page=7>
- Garrido, Á. (2020). *Los avances de la inteligencia artificial*. DYKINSON, S L.
- Gedeon, J., Brandherm, F., Egert, R., Grube, T., & Mühlhäuser, M. (2019). What the Fog? Edge Computing Revisited: Promises, Applications and Future Challenges. In *IEEE Access* (Vol. 7). <https://doi.org/10.1109/ACCESS.2019.2948399>
- Gil, M. (2019). *Azul de lactofenol: características, composición, preparación, usos*. 30 de Octubre. <https://www.lifeder.com/azul-de-lactofenol/>
- Gómez, T. (2017). Evaluación De Diferentes Medios De Cultivo Y Condiciones Para La Producción De Conidios De *Trichoderma* Spp Mediante Fermentación En Líquido Y

- Sólido. *Instituto Del Café de Costa Rica*, 15. [https://repositoriotec.tec.ac.cr/bitstream/handle/2238/7079/evaluacion\\_diferentes\\_medios\\_condiciones\\_produccion.pdf?sequence=1&isAllowed=y](https://repositoriotec.tec.ac.cr/bitstream/handle/2238/7079/evaluacion_diferentes_medios_condiciones_produccion.pdf?sequence=1&isAllowed=y)
- González, J. L. N. (2018). *Detección de incendios mediante identificación de humo con visión artificial en condiciones de iluminación variable*. <http://hdl.handle.net/20.500.11799/95189>
- Gutierrez, M., & Alayo, E. (2017). Degradación del insecticida alfacipermetrina por *Trichoderma harzianum*. *Véritas*, 16(2015), 61–64.
- Hamet, P., & Tremblay, J. (2017). Artificial intelligence in medicine. *Metabolism: Clinical and Experimental*, 69, S36–S40. <https://doi.org/10.1016/j.metabol.2017.01.011>
- Herrera Beltran, G. M. (2021). *Fermentación sólida en la industria alimentaria*. Trabajo. UNIVERSIDAD TÉCNICA DE AMBATO.
- Hispaviacion. (2020). *Los drones de búsqueda y rescate en Canadá utilizarán Inteligencia Artificial para optimizar las operaciones*. <https://www.hispaviacion.es/los-drones-de-busqueda-y-rescate-en-canada-utilizaran-inteligencia-artificial-para-optimizar-las-operaciones/>
- Infante, D., Martínez, B., Gonzáles, N., & Reyes, Y. (2019). MECANISMOS DE ACCIÓN DE *Trichoderma* FRENTE A HONGOS FITOPATÓGENOS. *Revista de Preservación Vegetal*, 24, 14–21.
- IOTHOOK. (n.d.). *Open Source IOT Dashboard*. Retrieved January 21, 2022, from <https://iothook.com/en/>
- ISO/IEC/IEEE. (2011). ISO/IEC/IEEE 42010. *Functional Analysis and Its Applications*, 16(4), 319–321. <https://doi.org/10.1007/BF01077867>
- Ivorra Martínez, E. (2021). *Introducción al software de visión artificial Sherlock*. <https://riunet.upv.es/handle/10251/165939>
- Jacas, J., Caballero, P., & Avilla, J. (2014). *El control biológico de plagas y enfermedades : la sostenibilidad de la agricultura mediterránea*. Universitat Jaume I. Servei de Comunicació i Publicacions. <https://elibro.net/es/ereader/utnorte/53255?page=122>.
- Lee, V. H., Hew, J. J., Leong, L. Y., Tan, G. W. H., & Ooi, K. B. (2020). Wearable payment:

- A deep learning-based dual-stage SEM-ANN analysis. *Expert Systems with Applications*, 157, 113477. <https://doi.org/10.1016/J.ESWA.2020.113477>
- López de Mántaras Badia, R., & Meseguer González, P. (2017). *Inteligencia artificial*. Editorial CSIC Consejo Superior de Investigaciones Científicas.
- Madhavan, S., & Jones, T. (2017, September 8). Deep learning architectures . *IBM Developer*.
- Malik, M. I. (2018). Google Colaboratory como alternativa para el procesamiento de una red neuronal convolucional. *International Journal of Advanced Research in Computer Science*, 9(2), 379–384. <https://doi.org/10.26483/IJARCS.V9I2.5760>
- Maliza, R., & Elena, R. (2017). “Sistema De Reconocimiento Facial Con Visión Artificial Para Apoyar Al Ecu-911 Con La Identificación De Personas En La Lista De Los Más Buscados.” *Repo.Uta.Edu.Ec*, 130. <http://repo.uta.edu.ec/bitstream/handle/123456789/5301/Mg.DCEv.Ed.1859.pdf?sequence=3>
- Marroquin, P. E. F., Leon, E. F. P., & Figueroa, F. A. V. (2004). *Sistema De Inspeccion Basado En Vision Artificial Aplicado a Un Proceso Industrial*.
- Martinez, J. (2020). *Gradiente Descendiente para aprendizaje automático*. IArtificial.Net. <https://www.iartificial.net/gradiente-descendiente-para-aprendizaje-automatico/>
- Maya Álvarez, M. A. (2017). *Caracterización y control de plagas en áreas edificadas y ajardinadas*. SEAG0110. IC Editorial.
- Militante, S. V., Gerardo, B. D., & Dionisio, N. V. (2019). Plant Leaf Detection and Disease Recognition using Deep Learning. *2019 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE)*, 579–582. <https://doi.org/10.1109/ECICE47484.2019.8942686>
- Ministerio del Trabajo del Ecuador. (2021). Salarios Mínimos Sectoriales 2021. In <https://www.trabajo.gob.ec/wp-content/uploads/2020/12/ANEXO-1-Estructuras-ocupacionales---salarios-m%C3%ADnimos-sectoriales-y-tarifas-sa.pdf?x42051> (Issue 1).
- Molnar, C. (2020). *Interpretable machine learning*. <https://books.google.com.ec/books?hl=es&lr&id=jBm3DwAAQBAJ&oi=fnd&pg=PP1&dq=machine+learning&ots=EguSXRCHYZ&sig=MIvEaFSqN2XGSjJONR7ylLgXKN>

I&redir\_esc=y&pli=1#v=onepage&q&f=false

Montero, B., Cevallos, H., & Dávila, J. (2018). Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software. *Espirales Revista Multidisciplinaria de Investigación* ISSN: 2550-6862, 2(17), 114–121. [https://www.researchgate.net/publication/327537074\\_Metodologias\\_agiles\\_frente\\_a\\_las\\_tradicionales\\_en\\_el\\_proceso\\_de\\_desarrollo\\_de\\_software](https://www.researchgate.net/publication/327537074_Metodologias_agiles_frente_a_las_tradicionales_en_el_proceso_de_desarrollo_de_software)

Muñoz Cuaical, S. D. (2016). Evaluación de la eficacia de *Trichoderma* sp y *Pseudomona* sp para biorremediación de suelos contaminados con hidrocarburos. *Universidad Politécnica Salesiana Sede-Quito*, 125.

Naranjo Silva, J. L. (2016). *DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA PARA LA CONTABILIZACIÓN AUTOMÁTICA DE UNIDADES FORMADORAS DE COLONIAS BACTERIANAS: COLIFORMES, MESÓFILOS AEROBIOS, BACILOS, ESTAFILOCOS Y LEVADURAS USANDO TÉCNICAS DE VISIÓN ARTIFICIAL* [Universidad de las Fuerzas Armadas (ESPE)]. <http://repositorio.espe.edu.ec/bitstream/21000/12017/1/T-ESPE-053357.pdf>

Nedumaan, J., Binford, T., Lepika, J., Tisa, J., Ruby, J., & Kumar, J. (2019). *Modern Deep Learning and Advanced Computer Vision: A Perspective Approach* (I. Published (ed.)). [https://books.google.com.ec/books/about/Modern\\_Deep\\_Learning\\_and\\_Advanced\\_Comput.html?id=70q8zAEACAAJ&redir\\_esc=y](https://books.google.com.ec/books/about/Modern_Deep_Learning_and_Advanced_Comput.html?id=70q8zAEACAAJ&redir_esc=y)

Núñez Sánchez, F. J. (2016). *Diseño de un sistema de reconocimiento automático de matrículas de vehículos mediante una red neuronal convolucional*. <http://openaccess.uoc.edu/webapps/o2/handle/10609/52222>

Oliveira, R. (2019). Hemograma - Como fazer e interpretar. In Amolca (Ed.), *LMP Editora* (Segunda).

OMS, O. M. de la salud, & FAO, O. acción de las N. U. para la A. y A. (2010). *Código internacional de conducta para la distribución y utilización de plaguicidas*.

Ordoñez, J. F., & Oñate, L. (2016). Detector De Bordes De Imágenes Usando Un Microcontrolador Arm. *Ingenius*, 16, 30. <https://doi.org/10.17163/ings.n16.2016.04>

Oropeza Clavel, C. A. (2007). *Modelado y Simulacion de un Sistema de Deteccion de Intrusos Utilizando Redes Neuronales Recurrentes*.

- Ortega Candel, J. M. (2018). *Hacking ético con herramientas Python*. RA-MA Editorial.  
<https://elibro.net/es/ereader/utnorte/106513?page=16>
- Pamoarato, M., Begonja, S., & Espósito, M. (2018). *Introducción a la química Hidrocarburos, alimentos y procesos industriales*. Primera Edición.  
<https://elibro.net/es/ereader/utnorte/138495>
- Pasquel, C. P. P. (2018). *Diseño De Un Sistema De Reconocimiento Automático De Vehículos Mediante El Uso De Redes Neuronales Profundas (Dnn)*. Universidad Técnica del Norte.
- Pastor, E. (2019). SomData, visión artificial para detectar los microorganismos del agua - Valencia Plaza. *Valencia Plaza*. <https://valenciaplaza.com/som-data-escaner-deteccion-microorganismos-agua>
- Patricia, J., Gamboa, Z., Alexandra, C., & Arreaga, L. (2018). Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software. Evolution of the Methodologies and Models used in Software Development. *INNOVA Research Journal*, 3(10), 20–33.
- Peña, A. R., & Sanchez, J. J. C. (2020). *Prototipo de geolocalización y reconocimiento aplicado a especies*. 89.
- Pinto Rodríguez, J. (2018). *Identificación de los productos y medios empleados para el control de plagas*. SEAG0110. 179.
- Polit, U., Val, C. D. E., & Catal, L. (2020). *Diseño de un proceso de producción de ácido acrílico , vía acroleína*. 1–71.
- Portero Donoso, P., & Mena Mena, B. (2017). “DESARROLLO DE UN PROTOTIPO PARA EL CONTROL DE CALIDAD DE LA CARNE BOVINA DETERMINADA POR SUS CARACTERÍSTICAS ORGANOLÉPTICAS, BASADO EN UN SISTEMA AUTOMÁTICO DE INSPECCIÓN POR VISIÓN ARTIFICIAL. ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO.
- Project Management Institute. (2017). *A guide to the project management body of knowledge PMBOK GUIDE* (Sexta). Project Management Institute, Inc.
- RAE. (2020). *Diccionario de la Real Academia de la Lengua Española*. Fundación La Caixa.  
<https://dle.rae.es/procesador?m=form>
- Rampérez Barrajon, J. (2021). El ácido poliláctico (PLA): Un material innovador en la

actualidad. *Universidad Almeriense*.

Raschka, S., Patterson, J., & Nolet, C. (2020). *information Machine Learning in Python: Main Developments and Technology Trends in Data Science, Machine Learning, and Artificial Intelligence*. <https://doi.org/10.3390/info11040193>

Reyes, N. (2017). *DISEÑO ÓPTIMO DE PLANTAS DE FERMENTACIÓN DISCONTINUA [UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA DEPARTAMENTO]*. <http://tinyurl.com/glmrldv>

Rodríguez, F. (2011). *Contacto – Bioseb Organics*. <https://www.bioseb.org/contacto>

Roman Bueno, J., & Gonzalez Mantilla, K. (2017). *Sistemas embebidos y Hardware libre*. <http://wiki.sc3.uis.edu.co/images/e/e1/GR7.pdf>

Sánchez Hernández, L., Rosa, A. M., Rosique Gil, J., & Pachero Figueroa, C. (2018). Diversidad del género *Trichoderma* (Hypocreae) en un Área Natural Protegida en Tabasco, México. *Acta Botanica Mexicana* 123, 167–182.

Sánchez Salinas, E., & Ortiz Hernández, L. (2018). Riesgos y estrategias en el uso de plaguicidas. *Inventio, La Génesis de La Cultura Universitaria En Morelos*, 11(14), 21–27.

Sandoval, M. C., & Belesansky, C. (2020). *Producción artesanal del hongo antagónico Trichoderma Persoon en sustrato sólido*. 7(3), 55–64.

Sanipatín, P. (2021). *RECONOCIMIENTO DE EXPRESIONES FACIALES A TRAVÉS DE UN ANÁLISIS DE PATRONES DE MOVIMIENTOS MUSCULARES FACIALES POR MEDIO DE TÉCNICAS DE APRENDIZAJE PROFUNDO*. Universidad Técnica del Norte.

Shi, W., & Dustdar, S. (2016). *The Promise of Edge Computing*. 18, Cloud Cover.

Silva, N., & Alvarez, S. (2013). Producción Artesanal y Control de Calidad del hongo antagonista *Trichoderma*. In *PRODUCCION ARTESANAL DE TRICHODERMA* (p. 48). [https://www.academia.edu/11343364/Producción\\_Artesanal\\_y\\_Control\\_de\\_Calidad\\_del\\_hongo\\_antagonista\\_Trichoderma](https://www.academia.edu/11343364/Producción_Artesanal_y_Control_de_Calidad_del_hongo_antagonista_Trichoderma)

Sittón-Candanedo, I., Alonso, R. S., Muñoz, L., & Rodríguez-González, S. (2019). Arquitecturas de Referencia Edge Computing para la Industria 4.0: una revisión.

*Arquitecturas de Referencia Edge Computing Para La Industria 4.0: Una Revisión. Edge Computing Reference Architectures for Industry 4.0: A Review, III Congreso Internacional en Inteligencia Ambiental, Ingeniería de Software y Salud Electrónica y Móvi*, 16–23. <https://rida2.utp.ac.pa/bitstream/handle/123456789/6471/Proceeding-AmITIC-2019-Pereira-Colombia.pdf?sequence=1&isAllowed=y>

Tecnologiavao. (2019). *El papel de la visión artificial en la industria automotriz*. <https://tecnologiaparalaindustria.com/el-papel-de-la-vision-artificial-en-la-industria-automotriz/>

Terou, H., & Parr, J. (1967). MICROORGANISMOS BENÉFICOS Y EFECTIVOS PARA UNA AGRICULTURA Y MEDIO AMBIENTE SOSTENIBLE. *Angewandte Chemie International Edition*, 6(11), 951–952. <https://itscv.edu.ec/wp-content/uploads/2018/10/MICROORGANISMOS-DEL-SUELO-PARA-LA-AGRICULTURA.pdf>

The MathWorks, I. (2022). *Thing Speak*. <https://thingspeak.com/>

Thew, R., Calarco, T., Roorda, P. D., Ca, O., Jones, K. P., Ca, K., Friesen, P., Ca, O., Patent, U. S., Ci, U. S., ITU, Systems, D., ITU-T, 129/Menkes/SK/II/2008, M. K. R. I. N. :, רגל, ר., Roberts, A., Mujtaba, A., Mujtaba, A., Mujtaba, A., ... Seifert, F. (2015). Estudio del adecuado crecimiento del hongo trichoderma harzianum y trichoderma hamatum en sustrato sólido [UNIVERSIDAD CENTRAL DEL ECUADOR]. In *Metrologia* (Vol. 53, Issue 5). [http://publicacoes.cardiol.br/portal/ijcs/portugues/2018/v3103/pdf/3103009.pdf%0Ahttp://www.scielo.org.co/scielo.php?script=sci\\_arttext&pid=S0121-75772018000200067&lng=en&tlng=en&SID=5BQIj3a2MLaWUV4OizE%0Ahttp://scielo.iec.pa.gov.br/scielo.php?script=sci\\_](http://publicacoes.cardiol.br/portal/ijcs/portugues/2018/v3103/pdf/3103009.pdf%0Ahttp://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0121-75772018000200067&lng=en&tlng=en&SID=5BQIj3a2MLaWUV4OizE%0Ahttp://scielo.iec.pa.gov.br/scielo.php?script=sci_)

Vázquez Nin, G. (2000). *Introducción a la Microscopía Electrónica Aplicada a Las Ciencias Biológicas*. Universidad México Americana del Norte. <https://books.google.es/books?hl=es&lr=&id=JNoMEWVRBmkC&oi=fnd&pg=PA7&dq=+MICROSCOPIA+&ots=CooOnaKOSD&sig=U0tn3vpeEjuWmdZJITZQLky8Zzo#v=onepage&q=MICROSCOPIA&f=false>

Vencoel. (2020, July 2). *Inteligencia aplicada a Sistemas de Visión Artificial: Evolución y Hardware - Venco Electronica Industrial*. <https://www.vencoel.com/inteligencia->

aplicada-a-sistemas-de-vision-artificial-evolucion-y-hardware/

- Woo, S. L., Ruocco, M., Vinale, F., Nigro, M., Marra, R., Lombardi, N., Pascale, A., Lanzuise, S., Manganiello, G., & Lorito, M. (2014). Trichoderma-based Products and their Widespread Use in Agriculture. *The Open Mycology Journal*, 8(1), 71–126. <https://doi.org/10.2174/1874437001408010071>
- Yordanov, Y. E., & Mladenov, V. M. (2018). Humanoid Robot Detecting Animals via Neural Network. *2018 14th Symposium on Neural Networks and Applications, NEUREL 2018*. <https://doi.org/10.1109/NEUREL.2018.8587017>