



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**Detección de cultivos infectados con tizón tardío en la papa (*solanum tuberosum*)  
basado en imágenes**

Andrade Basantes, Marco Vinicio

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica, Automatización y Control

Trabajo de titulación, previo a la obtención del título de Ingeniero en Electrónica,

Automatización y Control

PhD. Aguilar Castillo, Wilbert Geovanny

19 de agosto del 2023



Plagiarism report

Tesis-Andrade.docx

### Scan details

Scan time: August 17th, 2023 at 13:22 UTC      Total Pages: 74      Total Words: 18494

### Plagiarism Detection



| Types of plagiarism |      | Words |
|---------------------|------|-------|
| Identical           | 0.7% | 135   |
| Minor Changes       | 0.7% | 130   |
| Paraphrased         | 0%   | 0     |
| Omitted Words       | 3%   | 551   |

### AI Content Detection



Text coverage  
 AI text  
 Human text



WILBERT GEOVANNY AGUILAR CASTILLO

Firma:

**Dr. Wilbert G. Aguilar**

**Director de Tesis**



**Departamento de Eléctrica, Electrónica y Telecomunicaciones**

**Carrera de Ingeniería en Electrónica, Automatización y Control**

### **Certificación**

Certifico que el trabajo de titulación: **“Detección de cultivos infectados con tizón tardío en la papa (*solanum tuberosum*) basado en imágenes”** fue realizado por el señor **Andrade Basantes, Marco Vinicio**; el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

**Sangolquí, 19 agosto de 2023**



**PhD. Aguilar Castillo, Wilbert Geovanny**

C. C. 0703844696



**Departamento de Eléctrica, Electrónica y Telecomunicaciones**

**Carrera de Ingeniería en Electrónica, Automatización y Control**

**Responsabilidad de Autoría**

Yo, **Andrade Basantes, Marco Vinicio**, con cédula de ciudadanía n° 1717835746, declaro que el contenido, ideas y criterios del trabajo de titulación: **“Detección de cultivos infectados con tizón tardío en la papa (solanum tuberosum) basado en imágenes”** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

**Sangolquí, 19 de agosto de 2023**



.....  
**Andrade Basantes, Marco Vinicio**

C.C.: 1717835746



**Departamento de Eléctrica, Electrónica y Telecomunicaciones**

**Carrera de Ingeniería en Electrónica, Automatización y Control**

#### **Autorización de Publicación**

Yo, **Andrade Basantes, Marco Vinicio**, con cédula de ciudadanía n° 1717835746, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **“Detección de cultivos infectados con tizón tardío en la papa (solanum tuberosum) basado en imágenes”** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

**Sangolquí, 19 de agosto de 2023**



**Andrade Basantes, Marco Vinicio**

C.C.:1717835746

### **Dedicatoria**

*El siguiente trabajo se lo dedico a mis padres quienes me supieron apoyar a lo largo de este camino, con sabiduría, motivación y sobre todo paciencia. A mi hermana Samantha quien a pesar de ser pequeña y sin darse cuenta, me motivaba día a día con sus ocurrencias y alegrías. A mi abuelito Héctor que está en el cielo, quien siempre me ha enseñado a ser un hombre de bien. A mi abuelita Trinidad que ha sido como una madre, a mi hijo Davis que ha sido mi razón de superación.*

Marco Vinicio, Andrade Basantes

## **Agradecimiento**

*Quiero agradecer profundamente a mis padres quienes a pesar de las circunstancias nunca han dejado de apoyarme de una manera u otra, a mi familia que han estado ahí cuando los necesitaba, a mis profesores, quienes me han transmitido su conocimiento de manera asertiva, a mis amigos de la universidad que apoyándonos hemos podido culminar esta etapa de la vida.*

Marco Vinicio, Andrade Basantes

## Índice de Contenidos

|   |           |
|---|-----------|
| <b>Resumen</b>  | <b>15</b> |
| <b>Abstract</b>   | <b>16</b> |
| <b>Capítulo I</b>   | <b>17</b> |
| <b>Introducción</b>   | <b>17</b> |
| <b>Antecedentes</b>   | <b>17</b> |
| Justificación e Importancia .....   | 20        |
| Alcance del Proyecto.....   | 22        |
| Objetivos .....   | 23        |
| <i>Objetivo General</i> .....   | 23        |
| <i>Objetivos Específicos</i> .....  | 23        |
| <b>Capítulo II</b>  | <b>24</b> |
| <b>Fundamentación Teórica</b>   | <b>24</b> |
| <b>La papa (<i>Solanum tuberosum</i>)</b>   | <b>24</b> |
| Características del cultivo de la papa .....  | 25        |
| Gestión de suelo y crecimiento de la papa .....   | 26        |
| Enfermedades nocivas para las papas ( <i>Solanum Tuberosum</i> ).....   | 28        |
| <b>Enfermedad de Tizón Tardío (<i>Phytophthora Infestans</i>) y detección mediante la inteligencia artificial</b> | <b>33</b> |
| La Inteligencia artificial (IA).....  | 34        |
| <i>Aprendizaje automático (Machine Learning, ML)</i> .....  | 38        |
| Clasificación de aprendizaje automático.....  | 39        |
| <i>Tensorflow-Framework para el aprendizaje automático</i> .....  | 39        |
| <i>Python</i> .....   | 41        |
| <i>Keras</i> .....  | 43        |
| <i>Diferencias entre keras y tensorflow</i> .....   | 43        |



|  |           |
|--|-----------|
| <i>El campo de la visión por computadora</i> .....           | 45        |
| Neurona artificial .....                                     | 46        |
| Anatomía de una Neurona Artificial .....                     | 46        |
| Redes neuronales artificiales (ANN) .....                    | 48        |
| Hiperparametros.....   | 49        |
| Reconocimiento de imagen, aprendizaje profundo .....         | 52        |
| <i>Métodos de extracción de características</i> .....        | 54        |
| Extractor de características HAAR .....                      | 56        |
| HOG (Histograma De Gradientes Orientados).....               | 58        |
| Las redes neuronales convolucionales (CNN).....              | 60        |
| Aprendizaje de una red neuronal convolucional .....          | 64        |
| Proceso de muestreo .....                                    | 69        |
| Otras arquitecturas para extracción de características. .... | 72        |
| <i>Métodos de clasificación</i> .....                        | 73        |
| Bosques aleatorios (Random Forest) .....                     | 74        |
| Maquinas soporte Vectorial (SVM) .....                       | 75        |
| Aprendizaje Profundo (Deep Learning DL) .....                | 78        |
| Otros métodos de clasificación.....                          | 80        |
| <b>Capítulo III</b> .....                                    | <b>81</b> |
| Metodología y desarrollo de la solución.....                 | 81        |
| Obtención de imágenes para la Base de Datos.....             | 81        |
| <i>PlantVillage</i> .....                                    | 82        |
| <i>ImageNet</i> .....  | 82        |
| <i>Kaggle</i> .....  | 83        |
| <i>Consideraciones para la DataBase</i> .....                | 83        |
| Preparación de datos .....                                   | 85        |

|   |            |
|---|------------|
| <i>Tamaño de las imágenes</i> .....   | 85         |
| <i>Canales RGB (Red Green Blue)</i> .....                                       | 86         |
| <i>Optimización de la memoria</i> .....   | 87         |
| <i>División de los datos de para el entrenamiento validación y prueba</i> ..... | 88         |
| <i>Normalización de los datos</i> .....   | 89         |
| <i>Aumento de datos (imágenes)</i> .....  | 89         |
| Entrenamiento.....  | 90         |
| <i>Capas de convolución</i> .....   | 90         |
| <i>Capas de reducción de dimensionalidad</i> .....                              | 91         |
| <i>Capa Flatten</i> .....   | 92         |
| <i>Capas Densas</i> .....   | 92         |
| <b>Capítulo IV</b>  | <b>93</b>  |
| Pruebas.....  | 93         |
| <i>Modelo sin aumento de datos</i> .....  | 93         |
| <i>Modelo con aumento de datos</i> .....  | 94         |
| <i>Comparación del entrenamiento con el número de neuronas y épocas</i> .       | 94         |
| Resultados .....  | 96         |
| <i>Primera predicción</i> .....   | 96         |
| <i>Segunda predicción</i> .....   | 97         |
| Confianza del modelo.....   | 98         |
| <b>Capítulo V</b>   | <b>100</b> |
| Conclusiones.....   | 100        |
| Recomendaciones.....  | 100        |
| <b>Referencias Bibliográficas</b>   | <b>101</b> |

**Índice de tablas**

|  |    |
|--|----|
| Tabla 1 Número de imágenes y clases .....                    | 85 |
| Tabla 2 Porcentajes de división de datos para el modelo..... | 88 |

## Índice de figuras

|   |    |
|---|----|
| Figura 1 La papa.....   | 24 |
| Figura 2 Beneficios de consumir la papa.....  | 26 |
| Figura 3 Etapas de crecimiento de la papa .....   | 28 |
| Figura 4 Virus de enrollamiento de la hoja de papa .....  | 29 |
| Figura 5 Tizón tardío en la hoja de la papa .....   | 30 |
| Figura 6 Papa con daños causados por el virus mosaico.....  | 31 |
| Figura 7 Papa con daños causados por pudrición blanda bacteriana.....                                     | 31 |
| Figura 8 Planta de papa con la enfermedad de Marchitez bacteriana .....                                   | 32 |
| Figura 9 Detección de tizón tardío con técnicas de visión artificial e inteligencia artificial.....       | 34 |
| Figura 10 Lazo entre el aprendizaje automático y profundo con la inteligencia artificial .....            | 37 |
| Figura 11 Visión reducida del aprendizaje automático .....  | 38 |
| Figura 12 Diversas aplicaciones del aprendizaje automático .....  | 38 |
| Figura 13 Adaptación estructural de TensorFlow .....  | 40 |
| Figura 14 Python.....   | 42 |
| Figura 15 Keras vs. tf.keras .....  | 44 |
| Figura 16 Visión artificial en el campo de la Inteligencia artificial .....                               | 45 |
| Figura 17 Izquierda: La anatomía de una neurona biológica; Derecha: Modelo matemático de una neurona..... | 47 |
| Figura 18 Estructura de una Red neuronal conectada .....  | 48 |
| Figura 19 Visualización y representación matemática de diversas funciones de activación. ....             | 52 |
| Figura 20 Proceso de extracción de características de una imagen.....                                     | 54 |
| Figura 21 Izq: Filtros Haar, Der: Filtros Haar escalados en diferentes posiciones de la imagen.<br>.....  | 57 |
| Figura 22 Proceso de extracción de características de por HOG .....                                       | 59 |
| Figura 23 Extracción de características por redes neuronales convolucionales .....                        | 60 |

|   |    |
|---|----|
| Figura 24 Reconfiguración de red neuronal convolucional .....   | 62 |
| Figura 25 Lado izquierdo: Fotografía en tonos de gris para ser procesada; Lado derecho: Fotografía transformada en una matriz de píxeles. ....                    | 65 |
| Figura 26 Lado izquierdo: Imagen a color a ser procesada; Lado derecho: Imagen transformada en tres matrices de píxeles correspondientes a su escala en RGB. .... | 66 |
| Figura 27 Izquierda: Imagen de a procesar; Derecha: aplicación del Kernel o filtro. ....  | 67 |
| Figura 28 Recorrido del kernel en la imagen extrayendo características.....   | 67 |
| Figura 29 Implementación de la función ReLu .....   | 68 |
| Figura 30 Implementación del muestreo o subsampling y Max-Pooling .....   | 69 |
| Figura 31 Primera convolución.....  | 70 |
| Figura 32 Integración a una red convencional.....   | 71 |
| Figura 33 Deep Learning en Visión Artificial.....   | 73 |
| Figura 34 Estructura de la red de bosques aleatorios formados por distintos arboles .....   | 74 |
| Figura 35 Izquierda: datos lineales a clasificar, derecha: datos ya clasificados de forma lineal.   | 76 |
| Figura 36 Izquierda: datos no linealmente separables, derecha: datos ya clasificados no lineal .....  | 77 |
| Figura 37 Concurso de clasificación de imágenes imageNET .....  | 79 |
| Figura 38 Actividades realizadas en el proyecto .....   | 81 |
| Figura 39 Logo de PlantVillage .....  | 82 |
| Figura 40 Logo Imagenet .....   | 82 |
| Figura 41 Logo de Kaggle.....   | 83 |
| Figura 42 Hojas infectadas con el tizón tardío.....   | 84 |
| Figura 43 Hojas infectadas con el tizón temprano.....   | 84 |
| Figura 44 Imagen de una hoja con Tizón tardío y sus dimensiones .....   | 86 |
| Figura 45 Representación de una imagen con valores en RGB .....   | 86 |
| Figura 46 Matriz de 3x4 de la base de datos.....  | 87 |
| Figura 47 Ejemplo visual de los tipos de aumento de datos .....   | 90 |

|   |    |
|---|----|
| Figura 48 Gráfica de la Activación ReLU.....  | 91 |
| Figura 49 Representación gráfica de la función MaxPooling2D .....                       | 91 |
| Figura 50 Función SoftMax: Activación para la clasificación .....                       | 92 |
| Figura 51 Precisión y pérdida del entrenamiento y validación sin aumento de datos ..... | 93 |
| Figura 52 Precisión y pérdida del entrenamiento y validación con aumento de datos ..... | 94 |
| Figura 53 Desempeño del modelo con 32 neuronas en la capa densa y 20 épocas .....       | 95 |
| Figura 54 Desempeño del modelo con 64 neuronas en la capa densa y 30 épocas .....       | 96 |
| Figura 55 Primer ejemplo aplicando el modelo de solución.....                           | 97 |
| Figura 56 Segunda ejemplo aplicando el modelo de solución .....                         | 98 |
| Figura 57 Conjunto de imágenes aplicando el algoritmo de solución .....                 | 99 |

## Resumen

El proyecto consiste en el desarrollo de un modelo de detección de la enfermedad tizón tardío en los cultivos de papa aprovechando la fuerza de la Inteligencia Artificial y el aprendizaje profundo, modelo al cual también se le entreno para detectar el tizón temprano ya que comparten formas, colores, y texturas parecidas. Mediante el uso de algoritmos de visión por computadora, se procesan imágenes de hojas de papa, enfocándose específicamente en identificar y segmentar posibles áreas de infección.

Se genera una base de datos de diferentes repositorios de uso libre, para posteriormente con TensorFlow, popular en ámbitos de aprendizaje profundo, poder construir y entrenar un modelo de red neuronal convolucional. Este tipo de modelo es especialmente eficiente en tareas de procesamiento y clasificación de imágenes.

Una vez procesada la base de datos, se implementan técnicas de aumento de datos, como rotaciones y reflejos, para robustecer el modelo frente a variaciones y garantizar que pueda detectar el tizón en diferentes condiciones y etapas de la enfermedad. Además, se divide la base de datos en conjuntos de entrenamiento, validación y prueba, permitiendo evaluar de forma precisa el rendimiento del modelo.

La arquitectura del modelo convolucional se compone de múltiples capas, incluyendo capas de convolución, reducción de dimensionalidad y capas densamente conectadas. Este diseño busca extraer características esenciales de las imágenes y usarlas para realizar una clasificación efectiva. Esta solución no solo ofrece una herramienta de diagnóstico precisa sino también una estrategia proactiva para la gestión y tratamiento de cultivos.

*Palabras clave:* Red Neuronal Convolucional, tensor flow, tizón tardío

## Abstract

The project involves the development of a detection model for the late blight disease in potato crops, harnessing the power of Artificial Intelligence and deep learning. This model was also trained to detect early blight as they share similar shapes, colors, and textures. Through computer vision algorithms, images of potato leaves are processed, focusing specifically on identifying and segmenting potential infection areas.

A database is generated from various open-source repositories, followed by the use of TensorFlow, renowned in deep learning realms, to construct and train a convolutional neural network (CNN) model. This type of model is especially proficient in image processing and classification tasks.

Once the database is processed, data augmentation techniques, such as rotations and reflections, are implemented to fortify the model against variations and ensure its capability to detect blight under various conditions and disease stages. Furthermore, the database is divided into training, validation, and testing sets, allowing for accurate evaluation of the model's performance.

The architecture of the convolutional model comprises multiple layers, including convolutional layers, dimensionality reduction, and densely connected layers. This design aims to extract essential features from the images and employ them for effective classification. This solution not only provides an accurate diagnostic tool but also proffers a proactive strategy for crop management and treatment.

*Keywords:* Convolutional Neural Network, tensorflow, late blight



## Capítulo I

### Introducción

#### Antecedentes

En los últimos años la industria agrícola ha incursionado en el uso de tecnología para aumentar su productividad y eficiencia. Optando por el uso de herramientas pertenecientes a la industria 4.0, las cuales han convertido el mundo en un entorno en el cual para mantenerse competitivo respecto a los constantes cambios se ha vuelto indispensable automatizar procesos, monitorear bienes y analizar datos. Gracias al Machine Learning, la visión artificial y a su coexistencia con los vehículos aéreos no tripulados (UAVs por sus siglas en inglés) o también llamados drones, se han logrado implementar aplicaciones que han generado gran interés comercial como son: el monitoreo ambiental, monitoreo de desastres naturales (Cruz, Meneses, Aguilar, & Andrade-Miranda, 2019), vigilancia fronteriza, asistencia de emergencia, misiones de búsqueda y rescate, y comunicaciones de retransmisión (Yanmaz, Quaritsch, Yahyanejad, Rinner, Hellwagner, & Bettstetter, 2017).

En el país se han desarrollado prototipos orientados a áreas específicas como la agricultura, por ejemplo, se destaca el UAV/RPA que sobrevoló aproximadamente 200 hectáreas de terreno analizando su geografía a una altura de 180 metros, este dron junto a todas sus características cumplió sus objetivos con un alto grado de precisión ya que el recorrido había sido definido previamente por software (Pardo J. , 2016). Sistemas como el mencionado demuestran qué, si se equipa un UAV con herramientas como cámaras, equipos de transmisión y aditamentos extra se consigue gran versatilidad al momento de desarrollar una aplicación, la cual puede ser creada con diferentes algoritmos dependiendo de la decisión del programador.

Elegir un algoritmo es un paso crítico para el Machine Learning, según el tipo de respuesta esperada se pueden ejecutar diferentes técnicas de clasificación o regresión. La clasificación

brinda como resultado una clase entre un número limitado de clases, mientras que al usar regresión el resultado será un valor numérico (Pitarque, Roy, & Ruiz, 2018). Existen varias técnicas para cada tipo de método, en cuanto a la clasificación se tienen técnicas importantes como la regresión logística, que representa una red neuronal en miniatura; bosques aleatorios, Deep Learning, máquinas de vectores de soporte y árboles de decisión, que además pueden ser usadas para regresión. Adicionalmente, aparte de las técnicas mencionadas se usa con frecuencia la regresión lineal y no lineal (Carmona, 2014).

De manera general, existen casos donde se produce interacción entre clasificación y regresión, mejorando el proceso de reconocimiento y respuestas del computador. Dado que no se necesita solamente valores numéricos o juicios de valor, es necesario una combinación de los dos tipos de respuesta (Pitarque, Roy, & Ruiz, 2018).

Además, se necesitan técnicas para el reconocimiento de imágenes, dependiendo de la complejidad de estas, se elige que técnica usar; para imágenes básicas principalmente deben basarse en el color que presenten los píxeles con un rango específico, se asignan números a las diferentes tonalidades para que el computador pueda ejecutar las operaciones, generalmente se usa clasificadores con la técnica de regresión logística realizando el proceso iterativo. Sin embargo, si las imágenes resultan ser más complejas, una de las técnicas acertadas es la de redes neuronales. Cabe recalcar que cada píxel que se aprecia en una imagen es el resultado de la combinación de una serie de valores con funciones propias (Carmona, 2014).

La visión por computador ha revolucionado el campo científico, especialmente en la detección de objetos y procesamiento de imágenes; junto con el aprendizaje automático y el uso de algoritmos matemáticos se consigue la diferenciación de patrones. La extracción de

características y búsqueda de objetos con estas, son dos procesos sumamente importantes al momento de la detección (Kurtulmus & Kavdir, 2014).

Una de las principales técnicas es el Haar Feature, muy reconocida por ser la primera en ser usada para el reconocimiento facial en tiempo real, siendo su análisis de secuencias similar al de Fourier. El histograma de gradientes orientados HOG, representa otra técnica muy usada en este tipo de detecciones tomando en cuenta cambios de iluminación y contraste (Sharifara, A; Mohd Rahim, M; Anisi, Y, 2014).

Las redes neuronales han sido utilizadas para dar solución a una amplia variedad de tareas, como por ejemplo la visión por computador y el reconocimiento de voz, que poseen cierta dificultad para resolver haciendo uso de la ordinaria programación basado en reglas.

El hombre ha permanecido en constante búsqueda de recursos para mejorar su calidad de vida y facilitar sus obligaciones, de tal manera se ha conseguido el desarrollo de máquinas inteligentes y con esto el ingreso a las redes neuronales; la capacidad de memorizar y asociar hechos es propio de los humanos y trata de ser emulada por esta técnica expresada a través de algoritmos, llegando a ser unidades que intercambian información, reconocen patrones y repiten acciones simulando el proceso de aprendizaje (Quigley, Gerkey, Conley, Faust, Foote, & etal., 2010). Dentro de las redes neuronales artificiales se encuentran las FeedForward, que contienen flujo de información unidireccional, pero están interconectadas masivamente en paralelo y de forma jerárquica. Sin duda alguna, este tipo de redes neuronales representan un gran avance para la ciencia y junto con el uso de diferentes algoritmos han llegado a tener mucha importancia en la visión artificial (Carmona, 2014) (Chmaj & Selvaraj, 2015).

El Cloud Computing representa uno de los pilares en la cuarta revolución industrial, generando un gran impacto en la fabricación y en los diferentes tipos de industrias. Aunque la aceptación de la implementación de este tipo de tecnología ha sido tardía, debido a problemas de conectividad y de seguridad, los avances de la computación en la nube han permitido que

estas preocupaciones se desvanezcan. Los datos que se manejan se pueden compartir en un abrir y cerrar de ojos, reduciendo costos y aumentando productividad. Además, resulta de gran ayuda al momento de usar dispositivos de manera remota, reduciendo la carga que puede presentarse en el uso de baterías, mejorando la cobertura y brindando facilidades para la descarga de aplicaciones intensivas en computación. Siendo específicos: el procesamiento de voz y de imagen (Seongah, Simeone, & Joonhyuk, 2017).

### **Justificación e Importancia**

La papa de la misma manera que otro tipo de plantas se ve afectada por enfermedades de diferente origen que influyen negativamente en su desarrollo y rendimiento, se puede destacar enfermedades por plagas que repercuten tanto en su vida joven como en su vida adulta; entre las principales y notorias se encuentran: Tizón tardío de la papa (*Phytophthora infestans*), Pie negro o pudrición blanda (*Pectobacterium*), Sarna Común (*Streptomyces scabies*), Sarna Plateada (*Helminthosporium solani*), entre otras. Las cuales se caracterizan por generar una degradación notoria en los tejidos, que se observan en las partes externas de la planta, como su tallo, hojas y frutos (Méndez, 2012).

El tizón tardío es importante buscar minimizarla ya que afectado de gran manera en los cultivos del Ecuador (Tello, y otros, 2015). El patógeno tiende a ser muy agresivo y con las condiciones climáticas favorables las pérdidas alcanzan porcentajes elevados. Por esta razón el control de dicha enfermedad se le ha realizado con la aplicación de químicos en altas dosis debido a la resistencia que se va presentando.

La aplicación del control químico ha incrementado los costos de producción, así como daños a la salud tanto del agricultor, los consumidores y al medio ambiente (INIAP Estación Experimental Santa Catalina, 2015).

En los últimos años la utilización de herramientas tecnológicas en la agricultura ecuatoriana ha tenido una gran evolución, resaltando en gran ayuda para los empleados que

prestan su mano de obra, al punto de generar competitividad a nivel nacional. Existen diversos sistemas que se han aplicado a varios cultivos, entre los que se pueden mencionar: helicópteros fumigadores de arrozales, drones que verifican el suelo y planifican una organización adecuada de las plantas, y aplicaciones que notifican de algún problema en el sembrío (El Universo, 2019). Sin embargo, actualmente en el país no existen sistemas dedicados y capaces de detectar enfermedades en los cultivos de una manera anticipada, generando en el sector agrícola pérdidas de miles de hectáreas al año. La identificación rápida de enfermedades en los cultivos de papa o cual otro tipo podrían evitar pérdidas en gran medida, según Caref Sheldon, (Tipanluisa, 2019), se incrementaría la economía en un 15% del PIB en cinco años, produciendo solo lo necesario y limitando la sobre producción de cultivos que se pierden por las enfermedades. Además, se comenzaría a crear nuevas tecnologías adyacentes, logrando regenerar la fauna y ralentizando los impactos del efecto invernadero del calentamiento global.

A través del tiempo la globalización ha afectado a los procesos de producción, cambiándolos constantemente. Debido a factores como escasez de recursos y materia prima, impacto ambiental e incremento de edad de los trabajadores, las industrias han optado por la implementación de nuevas tecnologías de producción. La aparición de drones junto con la incesante búsqueda de innovación ha conseguido reducir pérdidas potenciales en diferentes sectores, sobre todo el sector agrícola (Chávez, 2018). A día de hoy con los avances tecnológicos se tiene la opción de atravesar enormes extensiones de sembríos, con UAV's que poseen cámaras que permiten captar y almacenar grandes cantidades de muestras, permitiendo brindar información la cual el ser humano no puede revelar a simple vista. El uso de UAVs junto con las TICs, manejo datos, y pilares de la industria 4.0 como el Machine Learning y el Cloud Computing han alcanzado el objetivo de que las industrias agrícolas obtengan información preventiva sobre las posibles enfermedades que afectarían a sus plantaciones (CEPAL & Rodrigues, 2015).

## **Alcance del Proyecto**

Se desarrollará un sistema que permita identificar la enfermedad tizón tardío en los cultivos de papa, utilizando técnicas de detección y clasificación pertenecientes a Machine Learning o Deep Learning, debido a la carencia de este tipo de sistemas en el país.

El trabajo se dividirá en las siguientes etapas:

La primera etapa será el estudio de la papa, gestión de suelos y la enfermedad tizón tardío que afecta a los cultivos de papa para lo cual se revisará la literatura respecto a los trabajos relacionados con la detección de enfermedades en cultivos de forma general y de forma específica en los que hayan abordado el problema del tizón tardío.

En la segunda etapa se realizará una revisión de las distintas bases de datos abiertas y disponibles en los repositorios institucionales de Universidades y/o Grupos de Investigación, con el fin de entrenar distintas propuestas algorítmicas basadas en: extracción de características; clasificadores; aprendizaje profundo. Nos permitirá identificar a cada tipo de imagen como, por ejemplo: color, forma, textura. En cada región de interés, el detector HAAR identifica las peculiaridades en un objeto, se basa en los niveles de intensidad que poseen los píxeles en una imagen dada, todo esto se desarrollara en con el lenguaje de programación Python, y los frameworks de inteligencia artificial como lo son, tensorflow y keras.

Además, el clasificador HOG (Histograma de Gradientes Orientados) usado en el contexto de computer vision y machine learning tiene la capacidad de extraer características, como forma y textura. Las redes neuronales convolucionales (CNN) tiene diversor aplicativos subcampo de la visión por computadora como es la detección, la clasificación y la segmentación objetos e imágenes. El principio de funcionamiento de una CNN es similar al de una red neuronal

convencional, pero con algunas diferencias clave. Entre otras técnicas que involucren Machine Learning y Deep Learning todo esto con el fin de elegir el que tenga menor coste computacional.

Finalmente, de desarrollaran pruebas del sistema a fin de mostrar su eficiencia, con las imágenes previamente obtenidas, se mostrará la imagen de las posibles áreas afectadas en una plantación, cabe recalcar que esto no será en tiempo real.

## **Objetivos**

### **Objetivo General**

Detectar los cultivos infectados con tizón tardío en la papa (*solanum tuberosum*) basado en imágenes.

### **Objetivos Específicos**

- Realizar la revisión de la literatura respecto a trabajos relacionados con técnicas de Machine Learning y Deep Learning para la detección en enfermedades de cultivos de papa.
- Realizar la construcción de una base de datos a partir de bases de datos ya existentes y de libre acceso en distintos repositorios de investigación destinada para el entrenamiento del algoritmo.
- Desarrollar las distintas propuestas algorítmicas para la prueba y análisis de los resultados con la base de datos generada.

## Capítulo II

### Fundamentación Teórica

#### La papa (*Solanum tuberosum*)

Es un tubérculo comestible que pertenece a la familia de las solanáceas. Es originario de las montañas de los Andes en Sudamérica, y se ha cultivado allí durante más de 8.000 años ver Figura 1. La papa es un cultivo alimenticio importante del mundo, se cree a día de hoy que es uno de los tres más importantes luego del arroz y del trigo (International Potato Center, 2023). Se produce en más de 100 países de todo el mundo, y es una fuente importante de alimento para millones de personas. Las papas son una buena fuente de carbohidratos, proteínas, fibra, vitaminas y minerales. Asimismo es una buena origen de antioxidantes, que pueden ayudar a proteger el cuerpo del daño causado por los radicales libres ver f. Las papas se pueden cocinar de diversas maneras, y se pueden usar en una variedad de platos (World Health Organization, 2022).

#### Figura 1

*La papa*



*Nota.* El gráfico muestra el vegetal denominado papa en Sudamérica, patata en usa y parte de Europa. Tomado del sitio web tuberculos.org | *La papa o patata*, (Leyva, 2023)



La papa es un cultivo resistente que puede tolerar una amplia gama de condiciones climáticas. También es un cultivo eficiente en el uso del agua, lo que lo convierte en una buena opción para los países con escasez de agua. La papa es una parte importante de la seguridad alimentaria mundial, y desempeña un papel importante en la nutrición de millones de personas (International Potato Center , 2023).

### **Características del cultivo de la papa**

Las temperaturas óptimas para el crecimiento de las raíces, el follaje y los estolones son relativamente bajas. Esto significa que las papas no son un cultivo ideal para las regiones cálidas y secas. Las papas también son un cultivo relativamente exigente en nutrientes (Journal of Soil Science and Plant Nutrition., 2015). Necesitan un suelo rico en materia orgánica y un buen suministro de nitrógeno, fósforo y potasio. Las papas también son sensibles a la falta de agua, por lo que es importante regarlas regularmente, se detalla a continuación algunas características del cultivo:

- La planta de la papa tiene un sistema radicular fibroso, lo que significa que tiene muchas raíces pequeñas y delgadas. Esto le permite absorber nutrientes y agua de una amplia área del suelo.
- Las papas pueden tolerar una amplia gama de condiciones climáticas, pero crecen mejor en climas templados con temperaturas entre 15 y 20 °C.
- Las papas necesitan un suelo rico en materia orgánica y un buen suministro de nitrógeno, fósforo y potasio.
- Las papas son sensibles a la falta de agua, por lo que es importante regarlas regularmente.
- Las papas son un cultivo relativamente resistente que puede tolerar una amplia gama de condiciones climáticas y plagas.

- Las papas son un cultivo importante que proporciona una fuente importante de alimento para millones de personas en todo el mundo.

Asu vez algunos beneficios de consumir la papa, son los siguientes:

Las papas son un alimento excelente para combatir enfermedades cardiovasculares y bajar la presión arterial, aliviar problemas de acné, dolores de cabeza, tal como se muestra en la Figura 2

**Figura 2**

*Beneficios de consumir la papa*



*Nota.* En la figura se puede observar los principales beneficios de consumir la papa, así como vitaminas que proporciona. Tomado del sitio web El comercio Peru.com | *La papa o patata y sus beneficios*, (El comercio-pe, 2023)

### **Gestión de suelo y crecimiento de la papa**

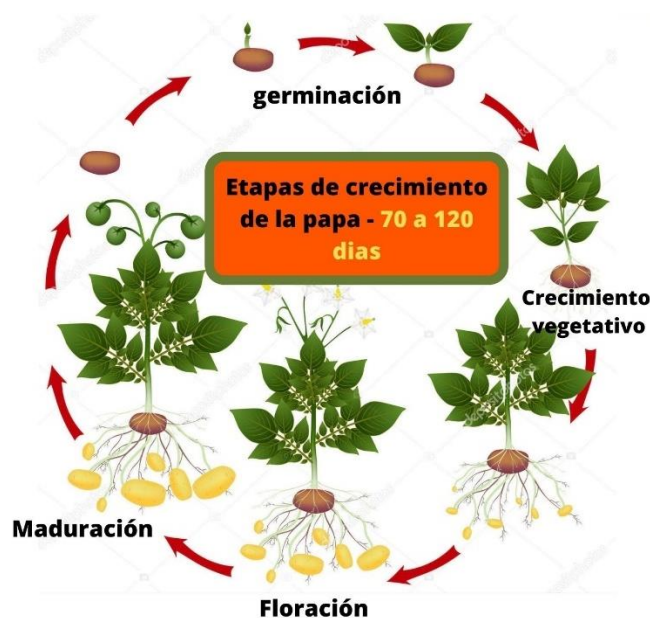
Para una mejor gestión de suelos (Abad, 2014) se detalle en breve los pasos a seguir para tener una buena cosecha de papas:

- **Elegir el suelo adecuado:** Las papas se pueden cultivar en una amplia gama de suelos, pero se desempeñan mejor en suelos ricos en materia orgánica y bien drenados. Los suelos arcillosos pesados no son ideales para el cultivo de papas porque son propensos a compactarse y retener demasiada agua. Los suelos arenosos ligeros también no son ideales para el cultivo de papas porque son propensos a secarse demasiado rápido.
- **Probar el suelo:** Antes de plantar papas, es importante probar el suelo para determinar el nivel de pH y los nutrientes disponibles. El pH del suelo debe estar entre 6,0 y 7,0 para el cultivo de papas. Si el pH del suelo es demasiado alto o demasiado bajo, es necesario agregar cal o azufre para ajustarlo. A su vez el suelo debe ser abundante en materia orgánica, tanto como estiércol o compost.
- **Aumentar materia orgánica:** La materia orgánica asiste a optimizar la estructura del suelo, lo cual ayudar favorece el desarrollo de las raíces de las papas. Permite retener la humedad en el suelo, motivo importante para las papas, que son sensibles a la sequía. La materia orgánica a su vez puede permite mejorar la salud del suelo en general y vigilar el crecimiento de las malas hierbas.
- **Fertilizar el suelo:** Las patatas necesitan muchos nutrientes para crecer sanas. El fertilizante más importante para las papas es el nitrógeno, que ayuda a las plantas a producir un follaje verde y saludable. Otros nutrientes importantes para las papas son el fósforo, el potasio, el magnesio y el calcio.
- **Cultivar el suelo:** El cultivo del suelo ayuda a eliminar las malas hierbas y a mantener el suelo suelto. Esto facilita el crecimiento de las raíces de las papas y ayuda a mejorar la aireación del suelo.
- **Regar las plantas:** Las papas necesitan mucha agua para crecer sanas. Es importante regar las plantas regularmente, especialmente durante las primeras etapas de crecimiento y durante períodos de sequía.

- **Recoger las papas:** Las papas se pueden cosechar cuando estén maduras, lo que suele ser entre 3 y 4 meses después de la siembra tal como se muestra en la **¡Error! No se encuentra el origen de la referencia..** Las papas se pueden cosechar m anualmente o con una excavadora de papas, dependiendo la economía del agricultor.

**Figura 3**

*Etapas de crecimiento de la papa*



*Nota.* En el grafico se aprecia las diferentes etapas de crecimiento de la papa. Tomado del sitio web [huertosymas.com](http://huertosymas.com) | *Etapas de crecimiento de la papa*, (Huertos y mas, 2023)

En muchos casos se ha notado que incluso las grandes empresas que tiene ambientes con rigurosos y estrictos cuidados no son objeto de excepción en cuanto a plagas y enfermedades que atacan a la planta se refiere a continuación se detalla algunas de las principales enfermedades que acechan a la papa (Food Security, 2018).

### **Enfermedades nocivas para las papas (*Solanum Tuberosum*)**

La papa es, desde hace muchos años, una de las principales fuentes de alimentación en muchas regiones del mundo y sus cosechas pueden verse afectadas por varias enfermedades,

plagas y además estaciones climáticas que afectan su cosecha, en este presente trabajo de titulación se estudiara algunas de las enfermedades entre las que destacan:

- Virus de enrollamiento de la hoja de la papa (Potato Leafroll Virus): Este virus se transmite por pulgones, como consecuencia de este virus las hojas de la planta se tornan de un color amarillento y el enrollamiento de las mismas tal como se muestra en la Figura 4, a su vez también provoca que la planta se debilite y que sus tubérculos reduzcan su rendimiento, aunque los síntomas pueden variar dependiendo del tipo de papa y las condiciones ambientales, incluso es posible que se pueda transmitir por el contacto con herramientas o plantas infectadas. Se considera una enfermedad realmente grave de la cual no se conoce cura actualmente.

#### Figura 4

*Virus de enrollamiento de la hoja de papa*



*Nota.* En la figura se puede apreciar a la izquierda una planta de papa afectada por la enfermedad del enrollamiento de la hoja, a la derecha una planta de papa sana. Tomado del sitio web [Agroavances.com](http://Agroavances.com) | *Papa: descifran el genoma del virus del enrollamiento de la hoja*, (Agrobiotecnología, 2017)

- Tizón Tardío (*Phytophthora Infestans*): Es una enfermedad fúngica propagada por esporas, las cuales pueden viajar por el aire o el agua para posteriormente adherirse a las plantas y germinar, es fácil de reconocer cuando una planta la posee por sus características manchas de moho oscuro tanto en sus hojas, tallos, así como en los tubérculos ver Figura 5. Las condiciones óptimas para su reproducción son un ambiente húmedo y caluroso, esto causa pérdidas considerables con los cultivos. Afecta también a sembríos de tomates y berenjenas, en pocos días puede llegar a destruir toda una cosecha y, aunque no hay cura para el tizón tardío, existen medidas y técnicas que se pueden aplicar para prevenir la enfermedad, frenarla o controlarla (Pérez & Forbes, 2008).

### Figura 5

*Tizón tardío en la hoja de la papa*



*Nota.* Se observa en el gráfico a la izquierda una planta de papa afectada por la enfermedad del *tizón tardío*, a la derecha una planta de papa que está en una etapa temprana de la enfermedad. Tomado del sitio web [portalfruticola.com](http://portalfruticola.com) | *Manejo Integrado del tizón tardío de la papa*, (Acuña I. , 2023)

- Virus del mosaico de la papa (Potato Mosaic Virus): Es un virus que causa deformaciones en los tubérculos y déficit en el crecimiento, reduce su rendimiento y provoca manchas claras y oscuras en las hojas de la planta, así como en el fruto en sí, tal como se muestra en la Figura 6. Es transmitido, en la mayoría de los casos, por pulgones que se alimentan de la savia de las plantas.

### Figura 6

*Papa con daños causados por el virus mosaico*



*Nota.* Se observa en la figura la afectación del virus mosaico en la papa de forma externa. Tomado del documento Protocolo de Diagnóstico Potato virus | *Enfermedad virus Mosaico en la papa*, (Dirección General de Sanidad, Centro de Nacional de Referencia Fitosanitaria, 2014)

- Pudrición blanda bacteriana (*Erwinia* spp.): Esta bacteria se encuentra por todo el mundo, ingresa a la planta por heridas o aberturas naturales, afecta directamente al tubérculo de la papa al destruir los tejidos y liberar toxinas. Produce manchas blandas, hundidas y despiden mal olor, los síntomas van a variar acorde a la progresión de la bacteria dentro de la planta y fruto verFigura 7. Puede propagarse a gran velocidad en condiciones de mucha humedad.

### Figura 7

*Papa con daños causados por pudrición blanda bacteriana*



*Nota.* Se observa en la figura la afectación de la pudrición blanda en la papa. Tomado del sitio web [portalfruticola.com](http://portalfruticola.com) | *Pudrición blanda de la papa: El patógeno que más afecta a este cultivo,* (Hortalizas, 2023)

- Marchitez bacteriana: (*Ralstonia solanacearum*): Produce afectaciones en el sistema vascular de la planta, conlleva a una marchitez progresiva que concluye en la muerte de la misma, la bacteria es caracterizada por transmitirse por el suelo, y es donde puede permanecer viva durante años ver Figura 8.

### Figura 8

*Planta de papa con la enfermedad de Marchitez bacteriana*





*Nota.* Se observa en la figura la afectación de la marchitez en la papa y algunos cultivos vegetales. Tomado del sitio web eorganic.org | *Marchitez o marchitamiento de la papa causada por Ralstonia solanacearum*, (Alam & Rustgi, 2023)

De todas las enfermedades mencionadas nos centraremos en la enfermedad que acecha a la papa como lo es Tizón Tardío, de la cual se ha propuesto dar una detección mediante técnicas vinculadas a la IA.

### **Enfermedad de Tizón Tardío (*Phytophthora Infestans*) y detección mediante la inteligencia artificial**

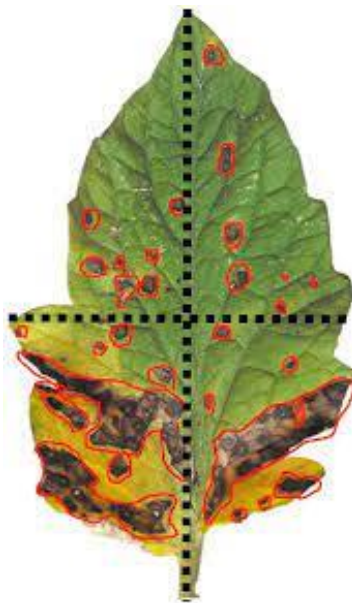
Detallando más de información con respecto a esta problemática, el tizón tardío es una de las enfermedades más devastadoras de la papa (Acuña, Sandoval, & Sepúlveda, 2021) y puede causar pérdidas de rendimiento de hasta el 100%. Los síntomas del tizón tardío incluyen manchas marrones en las hojas, tallos y tubérculos. Las hojas pueden marchitarse y morir, y los tubérculos pueden pudrirse (León, Arroyo, & Fuentes, 2019). El tizón tardío también puede causar la muerte de las plantas. Tomando como base en cuanto a las hojas se puede asociar con la inteligencia artificial mediante una serie de técnicas que pueden utilizarse para detectar el tizón tardío. Estas técnicas incluyen:

- *Análisis de imágenes:* Esta técnica utiliza imágenes de las plantas de papa para detectar signos de tizón tardío.
- *Modelos de aprendizaje automático:* Estos modelos pueden utilizarse para identificar los signos de tizón tardío en las imágenes de las plantas de papa (Costa, Goulart, & Fernandes, 2020).
- *Modelos de aprendizaje profundo:* Estos modelos pueden utilizarse para identificar los signos de tizón tardío en las imágenes de las plantas de papa con mayor precisión que los modelos de aprendizaje automático (Singh & Singh, 2021).

Las técnicas de IA pueden utilizarse para detectar el tizón tardío a una etapa temprana, lo que puede ayudar a los agricultores a tomar medidas para prevenir la propagación de la enfermedad tal como se muestra en la Figura 9.

### Figura 9

*Detección de tizón tardío con técnicas de visión artificial e inteligencia artificial*



*Nota.* En la figura se observa la detección de la enfermedad con técnicas de inteligencia artificial. Tomado del trabajo de titulación | *Algoritmo para la medición del grado de severidad de tizón tardío*, (Oliveros Sabogal, 2019).

### La Inteligencia artificial (IA)

La inteligencia artificial (IA) es una tendencia moderna en la informática que permite y se ocupa de la creación de sistemas inteligentes, procesos que les permiten razonar, administrar y desarrollar de forma independiente. La inteligencia artificial se puede dividir en dos categorías: inteligencia artificial estrecha e inteligencia artificial general. La Inteligencia artificial estrecha es un tipo de IA que está diseñada para realizar una tarea específica como conducir un automóvil

o jugar al ajedrez. La IA general es un tipo de IA que es capaz de realizar una amplia gama de tareas, como aprender, razonar y resolver problemas (Russell & Norvig, 2020).

La IA se utiliza en una amplia variedad de aplicaciones se detalla a continuación lo expuesto con los siguientes tópicos desarrollados

- **Robótica:** La robótica es el campo de la ingeniería que se encarga del boceto, construcción, trabajo, aplicación y también la reparación, así como mantenimiento de robots. Los robots son máquinas programables que pueden realizar tareas de forma autónoma o semiautónoma. La IA se utiliza en la robótica para controlar el movimiento de los robots, para detectar y evitar obstáculos, y para tomar decisiones sobre cómo completar tareas.
- **Reconocimiento de voz:** Es la capacidad de las máquinas para comprender y responder a las voces humanas. La inteligencia artificial se utiliza en el reconocimiento de voz para identificar a los hablantes, transcribir audio en texto y registrar dispositivos con la voz.
- **Procesamiento del lenguaje natural (PLN):** El PLN es el campo de la informática que se ocupa de la interacción entre las computadoras y el lenguaje humano natural. La IA se utiliza en el PLN para entender el significado del lenguaje humano, para generar lenguaje humano, y para traducir idiomas.
- **Ciencia de datos:** La ciencia de datos es el campo de la estadística y la informática que se ocupa de la extracción de conocimiento y significado de los datos (Sutton & Barto, 2018). La IA se utiliza en la ciencia de datos para desarrollar modelos de aprendizaje automático que pueden predecir resultados futuros, para identificar patrones en los datos, y para tomar decisiones informadas (Bishop, 2006).
- **Visión por computadora:** La visión por computadora es el campo de la ingeniería y la ciencia de la computación que se ocupa de la interpretación de imágenes y videos por parte de las computadoras (LeCun, Bengio, & Hinton, 2015). La IA con frecuencia es

relacionada al campo de la visión por computadora para reconocer objetos en fotos e imágenes, para perseguir el movimiento de objetos, así como también crear modelos que permitan construir diferentes objetos mediante las imágenes.

- Automatización robótica de procesos (RPA): La RPA es un tipo de automatización que utiliza software para automatizar tareas que normalmente son realizadas por humanos. La IA se utiliza en la RPA para desarrollar bots que pueden interactuar con sistemas informáticos y realizar tareas de forma autónoma.

Estos son solo algunos de los diferentes tipos de IA que existen. La IA es un campo en rápido desarrollo y nuevos tipos de IA se están desarrollando constantemente. La IA tiende a transformar muchas industrias y crear confort en la vida de las personas de muchas maneras. La IA está siendo usada en diversas aplicaciones, y se espera que se utilice en muchas más aplicaciones en el futuro.

Algunos de los beneficios potenciales de la IA incluyen:

- Mejora de la eficiencia: La IA se puede utilizar para automatizar tareas que actualmente son realizadas por humanos, lo que puede conducir a una mejora de la eficiencia.
- Mejora de la precisión: La IA se puede utilizar para hacer predicciones y tomar decisiones con mayor precisión que los humanos.
- Mejora de la seguridad: La IA se puede utilizar para detectar y prevenir amenazas, lo que puede conducir a una mejora de la seguridad.

Sin embargo, la IA también tiene algunos riesgos potenciales, como:

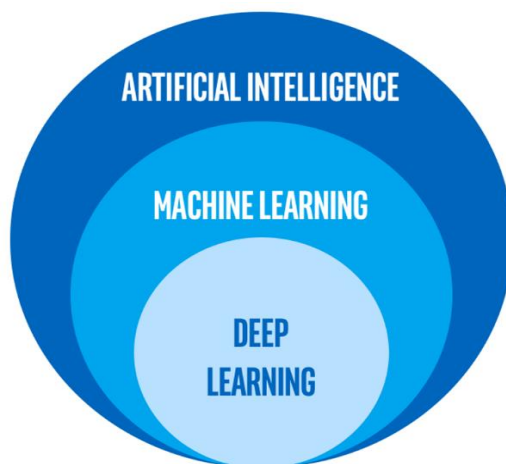
- Pérdida de puestos de trabajo: La IA se puede utilizar para automatizar tareas que actualmente son realizadas por humanos, lo que puede conducir a la pérdida de puestos de trabajo.
- Prejuicios: La IA puede ser sesgada, lo que puede conducir a decisiones injustas.

- Uso incorrecto: La IA puede ser utilizada para fines incorrectos, como el desarrollo de armas autónomas o la vigilancia masiva.

Es importante considerar los beneficios y los riesgos potenciales de la IA antes de utilizarla. Tal como se puede visualizar en Figura 10, como se unen todas las disciplinas de la IA brinda (Forsyth & Ponce , 2019), vinculadas en conjunto para dar soluciones a diversas aplicaciones sofisticadas, posteriormente se detalles, así como una explicación de la visión por computadora.

### Figura 10

*Lazo entre el aprendizaje automático y profundo con la inteligencia artificial*



*Nota.* Se puede observar en la figura los distintos tipos de aprendizaje que encuadra la inteligencia artificial. Tomado de *The Difference Between Artificial Intelligence, Machine Learning and Deep Learning*, (Robins, 2020).

La IA, el ML y el DL están estrechamente relacionados. La IA es el campo general, el ML es un subcampo que se ocupa de la creación de sistemas que pueden aprender, y el DL es un subcampo del ML que se ocupa del uso de redes neuronales artificiales.

## Aprendizaje automático (Machine Learning, ML)

El aprendizaje automático (ML) es un subcampo de la IA que se ocupa de la creación de sistemas que pueden aprender sin ser programados explícitamente.

### Figura 11

*Visión reducida del aprendizaje automático*



*Nota.* En la figura se aprecia como actúa un sistema de forma reducida de aprendizaje automático

Los datos y respuestas nos entregan características y dando como resultado un modelo pre-entrenado, como se muestra en la Figura 11 y estas condiciones se usan para extraer nuevas predicciones en los posteriores datos ingresados al sistema.

### Figura 12

*Diversas aplicaciones del aprendizaje automático*



*Nota* El gráfico representa las diversas aplicaciones del aprendizaje automático y su clasificación.

Tomado del sitio web trycore.co | *Aprendizaje automático para tu organización*, (Vargas, 2023)

## Clasificación de aprendizaje automático

Los sistemas de aprendizaje automático pueden aprender de los datos y mejorar su rendimiento con el tiempo. Tal como se observa en la Figura 12, hay muchos tipos diferentes de aprendizaje automático y aplicaciones del mismo, pero los más comunes son:

- *Aprendizaje supervisado*: En el aprendizaje supervisado, se le proporciona al sistema un conjunto de datos de entrenamiento que incluye entradas y salidas. El sistema aprende a relacionar las entradas con las salidas, Predice el futuro a partir de datos históricos. Es muy usado para solucionar inconvenientes de regresión y categorización (Murphy, 2012)
- *Aprendizaje no supervisado*: En el aprendizaje no supervisado, al sistema no se le proporciona una respuesta deseada para cada entrada. En cambio, el sistema debe aprender a encontrar patrones en los datos por sí mismo. Esto se puede hacer utilizando una variedad de métodos, como el agrupamiento, la reducción de dimensionalidad y la detección de anomalías. Los datos son bosquejados para desarrollar nuevos conocimientos por medio del descubrimiento de regularidades en los datos (MacQueen, 1967).
- *Aprendizaje por refuerzo*: es un tipo de aprendizaje automático en el que el sistema aprende a tomar decisiones a partir de la retroalimentación. El sistema es recompensado por tomar decisiones que conducen a resultados deseados y castigado por tomar decisiones que conducen a resultados no deseados. Es un aprendizaje interesante en los sistemas de IA, es debido a que al sistema no se le ingresa una gran cantidad de información (Vapnik, 1995).

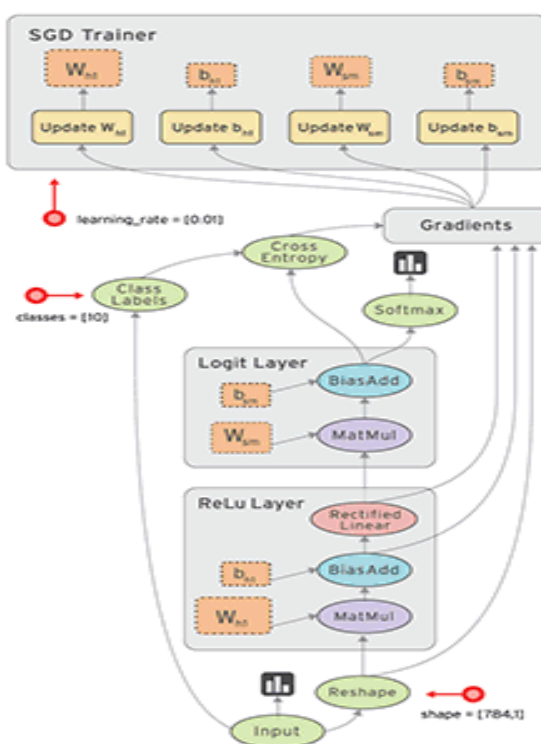
## Tensorflow-Framework para el aprendizaje automático

TensorFlow representa una poderosa herramienta de programación de gráficos computacionales de alto nivel, desarrollada por Google como una biblioteca de software de

código abierto y gratuita. Su enfoque principal radica en el campo del aprendizaje automático y el procesamiento del lenguaje natural. De esta manera, se convierte en un recurso fundamental tanto para la investigación como para la implementación de modelos avanzados de aprendizaje automático. Está diseñado para ser flexible y eficiente, y admite una amplia gama de tareas de aprendizaje automático, incluidas la clasificación, la regresión y la agrupación (Abadi, y otros, 2016).

**Figura 13**

*Adaptación estructural de TensorFlow*



*Nota.* En la imagen se observa como realiza TensorFlow los cálculos internos de una Red neuronal Artificial. Tomado de *Learn| TensorFlow 4: Convolutional Neural Network (CNN's)*, (Tenezaca, Canchignia, Mendoza, & etal., 2019)



TensorFlow se basa en el concepto de gráficos computacionales, que son un tipo de representación gráfica de datos y operaciones tal como se observa en la Figura 13. Los gráficos computacionales se utilizan para representar modelos de aprendizaje automático, y TensorFlow proporciona una serie de herramientas para crear, entrenar y evaluar modelos de aprendizaje automático utilizando gráficos computacionales.

Es una biblioteca poderosa y flexible que se puede utilizar para una amplia gama de tareas de aprendizaje automático, Esta herramienta es ampliamente utilizada para desarrollar modelos de aprendizaje automático y aprendizaje profundo, pudiendo ser implementados tanto en entornos locales como en la nube, aprovechando aceleradores de hardware como GPUs y plataformas de computación paralela para el procesamiento de tensores CUDA. También ofrece opciones para implementaciones en navegadores web o en dispositivos móviles.

Aquí hay algunos de los beneficios de usar TensorFlow:

- Es gratuito y de código abierto.
- Es flexible y eficiente.
- Es compatible con una amplia gama de tareas de aprendizaje automático.
- Tiene una gran comunidad de usuarios y desarrolladores.
- Posee compatibilidad con Linux, macOS y Windows.

## **Python**

Python es un lenguaje de programación de propósito general y alto nivel ampliamente conocido. Su característica distintiva radica en ser un lenguaje interpretado, lo que implica que no es necesario compilarlo previamente para ejecutarlo. Con una notable relevancia en el campo del aprendizaje automático y la investigación, Python goza de gran popularidad entre investigadores y desarrolladores de diversos rincones del planeta (van Rossum, 1991).

Las características clave de Python incluyen:

- Es un lenguaje de código abierto. Esto significa que es gratuito y cualquiera puede usarlo o modificarlo.
- Es un lenguaje fácil de aprender. Posee sintaxis sencilla y transparente, fácil de aprender para personas que no están familiarizadas con la programación.
- Es un lenguaje versátil. Se puede utilizar para una amplia gama de tareas, incluyendo el desarrollo web, el procesamiento de datos, la ciencia de los datos y el aprendizaje automático.
- Tiene una gran comunidad de usuarios y desarrolladores. Hay mucha documentación, tutoriales y soporte disponibles para Python.
- Python es un lenguaje muy adecuado para iniciar en temas como el aprendizaje automático y la investigación. Es un lenguaje versátil, fácil de aprender y tiene una gran comunidad. Por estas razones, Python es una buena opción para las personas que quieren aprender sobre el aprendizaje automático o desarrollar modelos de aprendizaje automático.

#### Figura 14

*Python*



*Nota.* En la imagen se observa el logo del lenguaje de programación Python. Tomado de *Python: Web Oficial*, (Python Sitio Web Oficial, 2021)

## **Keras**

Keras es una biblioteca de alto nivel de Python para redes neuronales profundas, construida sobre TensorFlow. Es una biblioteca liviana y fácil de usar, que hace que sea rápido y fácil crear y entrenar redes neuronales. Keras es una biblioteca popular para la investigación y el desarrollo de redes neuronales profundas, y es utilizada por investigadores y desarrolladores de todo el mundo (Chollet, Keras: The Python Deep Learning library, 2015).

Algunas de las características clave de Keras incluyen:

- Es una biblioteca de alto nivel. Esto significa que es fácil de usar y no requiere conocimientos profundos de TensorFlow.
- Es una biblioteca liviana. Esto significa que es rápido y eficiente, lo que es importante para el entrenamiento de redes neuronales profundas.
- Es una biblioteca flexible. Esto significa que se puede utilizar para una amplia gama de tareas de redes neuronales profundas.
- Tiene una gran comunidad de usuarios y desarrolladores. Esto significa que hay mucha documentación, tutoriales y soporte disponibles.

Keras es una biblioteca popular para la investigación y el desarrollo de redes neuronales profundas. Es una biblioteca liviana, fácil de usar y flexible, que hace que sea rápido y fácil crear y entrenar redes neuronales profundas.

## **Diferencias entre keras y tensorflow**

Keras y TensorFlow son dos frameworks de aprendizaje profundo de gran renombre, ampliamente reconocidos por su naturaleza de código abierto. Ambos se utilizan para construir y entrenar redes neuronales, pero tienen algunas diferencias clave.

- Keras es un marco de alto nivel, lo que significa que es más fácil de usar que TensorFlow. Keras también es más flexible que TensorFlow, ya que puede

utilizarse con diferentes backends, como TensorFlow, CNTK y Theano (Chollet, Keras, 2015).

- TensorFlow se caracteriza por ser un framework de bajo nivel, lo que implica que brinda un mayor grado de control en el proceso de aprendizaje profundo.. TensorFlow también es más rápido que Keras, lo que lo hace ideal para aplicaciones que requieren un alto rendimiento (Abood, Al-Rubaiee, & Khasawneh, 2019).

En general, Keras es una buena opción para los principiantes en aprendizaje profundo, mientras que TensorFlow es una buena opción para los usuarios más avanzados que necesitan un control más preciso sobre el proceso de aprendizaje profundo tal como se puede observar en la Figura 15, en donde la entrada de la red neuronal artificial es más sencillo ingresar los datos con keras, pero para tratar y extraer las características con mucho más precisas se usa tensorflow.

### Figura 15

*Keras vs. tf.keras*



*Nota.* En la imagen se observa una pequeña red neuronal artificial entre keras y tensorflow.

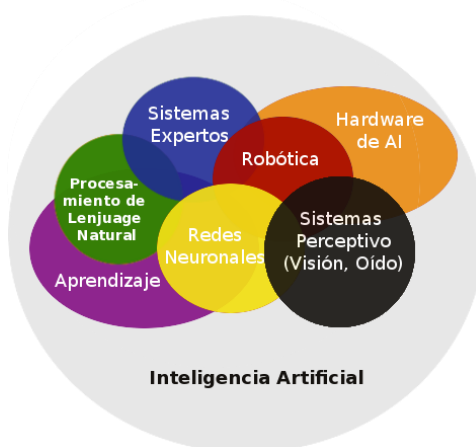
Tomado de *Learn| Deep learning with Keras*, (Chollet, Keras, 2015)

## El campo de la visión por computadora

Dentro del contexto del presente proyecto de graduación, uno de los componentes clave en los aplicativos de IA es la visión por computadora. Esta área en particular se encuentra representada en la Figura 16, donde se destaca su enfoque en la intersección entre el aprendizaje automático y profundo.

### Figura 16

*Visión artificial en el campo de la Inteligencia artificial*



*Nota.* El gráfico muestra los diversos aplicativos se encuentran en la Inteligencia artificial así como la visión artificial. Tomado de *¿Cómo iniciarse en la inteligencia artificial?*, (Labra, 2021)

La visión artificial y la visión por computadora citados en (Forsyth & Ponce , 2019) y (Gonzalez & Woods , 2018) respectivamente, son dos campos estrechamente relacionados, pero no son lo mismo. Como tal la visión artificial es el estudio de cómo los sistemas informáticos pueden procesar información visual del mundo real. La visión por computadora es el uso de técnicas de procesamiento de datos para analizar imágenes y videos (Szeliski , 2010). La visión artificial a menudo se utiliza para aplicaciones que requieren la comprensión de la forma, el movimiento y la estructura de los objetos en el mundo real (Forsyth & Ponce , 2019). Por ejemplo, la visión artificial se utiliza en sistemas de seguridad para detectar intrusos, en sistemas de conducción autónoma para navegar por la carretera y en sistemas de reconocimiento facial para

identificar personas, en cambio la visión por computadora a menudo se utiliza para aplicaciones que requieren la extracción de información de imágenes y videos. Por ejemplo, la visión por computadora se utiliza en sistemas de reconocimiento de objetos para identificar objetos en una imagen, en sistemas de seguimiento de movimiento para rastrear el movimiento de objetos en un video y en sistemas de clasificación de imágenes para clasificar imágenes en categorías.

En general, la visión artificial se centra en la comprensión del mundo real, mientras que la visión por computadora se centra en la extracción de información de imágenes y videos. A continuación, se realiza una breve descripción de algunos conceptos importantes antes del ingreso a los métodos de acción que se aplicaran para lograr los objetivos planteados en el desarrollo de este trabajo de titulación

## **Neurona artificial**

Las redes neuronales artificiales son uno de los desarrollos más significativos en IA. Antes de definir este concepto, se aclara que una neurona artificial es lo que representa, una neurona artificial es un componente que intenta imitar el comportamiento de una neurona biológica que se encuentra en el cerebro humano ver Figura 17. Las neuronas artificiales intentan conectarse entre sí para construir redes neuronales que puedan almacenar y procesar información.

## **Anatomía de una Neurona Artificial**

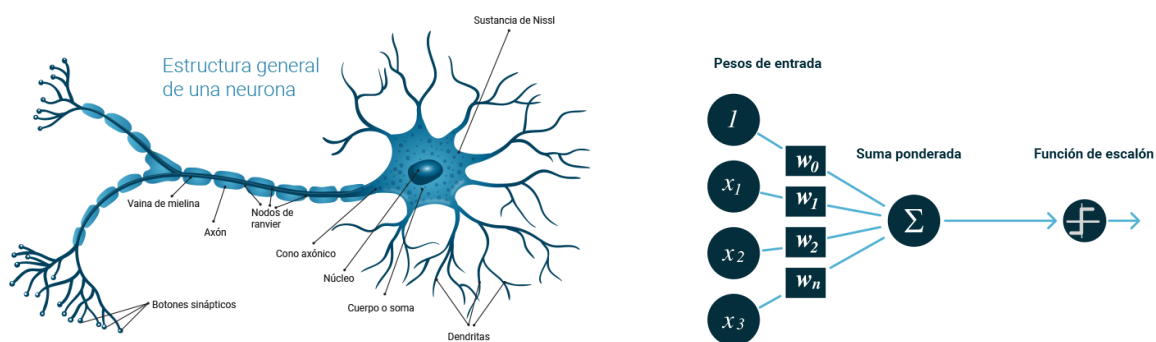
Las neuronas artificiales han sido estudiadas por muchos años, y una de las primeras investigaciones fue la de McCulloch y Pitts en 1943. McCulloch y Pitts propusieron un modelo matemático de una neurona artificial que era muy similar al de una neurona biológica. En 1958, Frank Rosenblatt propuso el perceptrón, que es un tipo de neurona artificial que puede aprender a realizar tareas simples. El perceptrón está constituido por una operación matemática de suma, donde cada una de sus entradas se representa como datos de entrada multiplicados por un valor

denominado peso sináptico. Los pesos sinápticos controlan la importancia de cada entrada, y son ajustados por el perceptrón a través del proceso de aprendizaje.

El perceptrón tuvo un impacto significativo en el desarrollo de las redes neuronales artificiales, y se utilizó para resolver una variedad de problemas, incluyendo el reconocimiento de patrones y la clasificación. Sin embargo, el perceptrón solo puede aprender a realizar tareas que pueden ser divididas en funciones lineales. En los años 1960, se desarrollaron redes neuronales artificiales más complejas que podían aprender a realizar tareas que no eran lineales. Estas redes neuronales artificiales se conocen como redes neuronales no lineales, y se utilizan en una variedad de aplicaciones, el reconocimiento de imágenes, el procesamiento del lenguaje natural y otras áreas donde el aprendizaje automático ha demostrado ser altamente efectivo.

### Figura 17

*Izquierda: La anatomía de una neurona biológica; Derecha: Modelo matemático de una neurona.*



*Nota.* En la imagen se presenta una comparativa entre una neurona biológica y una neurona artificial representada de manera matemática. Tomada de *Inspiración biológica de las redes neuronales artificiales*, (Campos Soberanis, 2018)

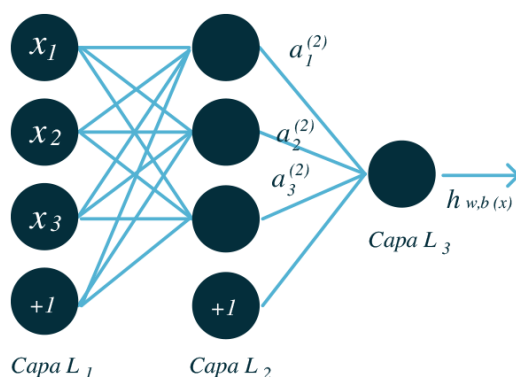
## Redes neuronales artificiales (ANN)

Una red neuronal artificial (RNA) es un modelo de aprendizaje automático inspirado en el sistema nervioso biológico. Las neuronas artificiales son unidades que se conectan entre sí para formar una red. Las neuronas artificiales reciben entradas, procesan estas entradas y producen una salida. Los pesos de las conexiones entre las neuronas artificiales se ajustan durante el proceso de aprendizaje (Haykin, 1994).

Las neuronas artificiales se conectan entre sí para formar capas neuronales. Cada capa neuronal recibe entradas de la capa anterior y produce salidas que se envían a la capa siguiente tal como se muestra en la Figura 18. La última capa neuronal produce una salida que es la predicción para la entrada dada. Todas las neuronas en una capa trabajan de forma independiente y no comparten información alguna. Esto significa que cada neurona en una capa procesa la información de forma independiente y produce su propia salida (Hinton & Sejnowski, 1986). La unión de capas neuronales permite a las redes neuronales artificiales aprender tareas complejas. Por ejemplo, una red neuronal artificial puede aprender a reconocer caras, identificar objetos o traducir idiomas (Nielsen, 2015).

**Figura 18**

*Estructura de una Red neuronal conectada*





*Nota.* En la figura se puede apreciar el diseño de una red neuronal totalmente conectada.. Tomada del sitio web | *Inspiración biológica de las redes neuronales artificiales*, (Campos Soberanis, 2018)

### **Hiperparámetros.**

Son parámetros que controlan el rendimiento de una red neuronal. Algunos de los hiperparámetros más importantes son (Rodríguez V. , 2018):

- **Tamaño de la red:** El tamaño de la red se define por la cantidad de neuronas presentes en cada capa. Una red más grande puede aprender tareas más complejas, pero también es más difícil de entrenar.
- **Número de capas:** Se refiere a la cantidad de capas de neuronas que conforman la red. Una red con más capas puede aprender funciones más complejas, pero también es más difícil de entrenar.
- **Tipo de algoritmo de entrenamiento:** El algoritmo de entrenamiento es el algoritmo que se utiliza para ajustar los pesos de la red. El algoritmo de entrenamiento afecta a la velocidad y la eficacia con la que la red aprende. Algunos algoritmos de entrenamiento son más adecuados para ciertas tareas que otros.
- **Función de pérdida:** Es una herramienta matemática empleada para cuantificar el error de la red neuronal. La función de pérdida afecta a la forma en que la red aprende. Algunas funciones de pérdida son más adecuadas para ciertas tareas que otras.
- **Tasa de aprendizaje:** La tasa de aprendizaje es el tamaño de los pasos que la red da al ajustar los pesos. La tasa de aprendizaje afecta a la velocidad con la que la red aprende. Una tasa de aprendizaje demasiado alta puede hacer que la red converja a un mínimo local, mientras que una tasa de aprendizaje demasiado baja puede hacer que la red tarde demasiado en converger.

- **Regularización:** Es una técnica empleada para prevenir que la red neuronal aprenda en exceso. La regularización afecta a la capacidad de la red para generalizar a nuevos datos. Algunos tipos de regularización son más adecuados para ciertas tareas que otros.

Otros conceptos validos en cuanto a Hiperparámetros se refiere se muestran a continuación:

- **Sobreajuste (Overfitting):** El sobreajuste ocurre cuando la red neuronal se adapta excesivamente a los datos de entrenamiento y pierde su capacidad para generalizar correctamente a nuevos datos. Esto puede ocurrir si la red es demasiado grande o si la tasa de aprendizaje es demasiado alta. Se sugiere de preferencia evitar en lo posible este parámetro.
- **Bajo ajuste (Underfitting):** El subajuste ocurre cuando la red neuronal no es capaz de aprender los patrones de los datos de entrenamiento. Esto puede ocurrir si la red es demasiado pequeña o si la tasa de aprendizaje es demasiado baja.
- **Datos de entrenamiento:** Las muestras, conocidas como datos de entrada, son esenciales para el aprendizaje de la red neuronal, ya que permiten extraer las características necesarias. Estos datos suelen ser bastante extensos y se dividen en dos conjuntos: aproximadamente del 70% al 80% se destina al entrenamiento, mientras que el resto se reserva para la validación. De esta manera, la red puede aprender y generalizar eficazmente a partir de las muestras proporcionadas.
- **Datos de validación:** son un conjunto de datos que se separa del conjunto de entrenamiento y no se utilizan para entrenar la red neuronal. Se utilizan para evaluar el rendimiento de la red neuronal y para ajustar los hiperparámetros.
- **Retropropagación (Back-Propagation):** utilizado para ajustar los pesos de una red neuronal durante el proceso de entrenamiento. La retropropagación calcula el error de la red neuronal y luego utiliza este error para ajustar los pesos de la red neuronal.

- **Optimizador:** es un algoritmo que se emplea para ajustar los pesos de una red neuronal durante el proceso de entrenamiento
- **Taza de aprendizaje:** (Learning Rate, por sus siglas en inglés), es un parámetro crucial en el entrenamiento de una red neuronal que controla la velocidad a la que el modelo aprende a partir de los datos de entrenamiento. Una tasa de aprendizaje alta hará que la red aprenda rápidamente, pero también puede ser más propensa a sobreajuste. Una tasa de aprendizaje baja hará que la red aprenda más lentamente, pero también será más estable y menos propensa a sobreajuste.
- **Función de pérdida:** la función en cuestión evalúa la discrepancia entre las predicciones realizadas por una red neuronal y los valores reales de los datos. La función de pérdida se utiliza para entrenar la red neuronal, ajustando los pesos de la red de modo que la función de pérdida se minimice.
- **Época:** Una época es una iteración completa sobre el conjunto de datos de entrenamiento. En cada época, la red neuronal se alimenta con todos los datos de entrenamiento y se actualizan los pesos de la red.
- **Tamaño de lote:** (Batch Size. traducido del inglés), es el número de datos que se utilizan para actualizar los pesos de la red neuronal en cada época. Un tamaño de lote más grande aporta mayor estabilidad a la red neuronal durante el entrenamiento, pero también implica una mayor duración en el proceso. Por otro lado, un tamaño de lote más reducido acelerará el entrenamiento, pero podría aumentar el riesgo de sobreajuste. Además, en el caso de conjuntos de datos extensos, se requerirán más recursos computacionales. Sin embargo, al variar este parámetro a lotes más pequeños, se permite que la red neuronal aprenda más rápidamente, lo que disminuye el tiempo de entrenamiento.
- **Funciones de activación:** La función de activación es una función matemática que se aplica a la salida de cada neurona. La función de activación afecta a la forma en que la red aprende. Algunas funciones de activación son más adecuadas para ciertas tareas

que otras. Estas funciones de activaciones pueden ser de dos clases, funciones lineales y funciones no lineales tal como se muestra en Figura 19.

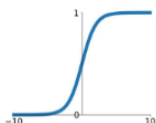
**Figura 19**

*Visualización y representación matemática de diversas funciones de activación.*

## Activation Functions

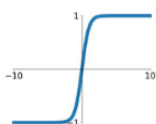
### Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



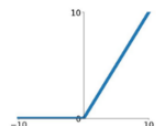
### tanh

$$\tanh(x)$$



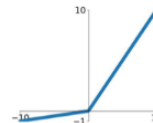
### ReLU

$$\max(0, x)$$



### Leaky ReLU

$$\max(0.1x, x)$$

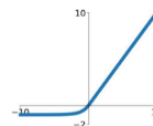


### Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

### ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



*Nota.* En la imagen se puede observar las distintas funciones de activación usadas en las Redes neuronales artificiales. Tomado del sitioweb [keepcoding.io](https://keepcoding.io) | ¿Qué es una función de activación en Deep Learning?, (keepcoding, 2022)

## Reconocimiento de imagen, aprendizaje profundo

El reconocimiento de imágenes se refiere a una tarea en la que se busca identificar objetos presentes en imágenes. El machine learning, una rama de la inteligencia artificial, proporciona la capacidad a las computadoras de aprender de datos sin requerir una programación explícita (Krizhevsky, Sutskever, & Hinton, 2012). La combinación del reconocimiento de imágenes con machine learning ha dado lugar a una potente herramienta que se aplica en diversas áreas, abriendo paso a una amplia gama de aplicaciones, las cuales son:

- *Clasificación de imágenes:* El reconocimiento de imágenes puede utilizarse para clasificar imágenes en categorías, como personas, animales, objetos u otros tipos de cosas.

- *Reconocimiento de caras*: El reconocimiento de caras puede utilizarse para identificar personas en imágenes.
- *Detección de objetos*: El reconocimiento de objetos puede utilizarse para identificar objetos en imágenes, como coches, personas, animales u otros objetos.
- *Seguimiento de objetos*: El seguimiento de objetos puede utilizarse para seguir el movimiento de objetos en imágenes.
- *Reconocimiento de texto*: El reconocimiento de texto puede utilizarse para extraer texto de imágenes.

El proceso de reconocer imágenes, según la aplicación puede tener un orden donde se puede seguir el siguiente patrón:

**Etiquetado de imágenes:** El etiquetado de imágenes es el proceso de asignar etiquetas a las imágenes. Las etiquetas pueden ser categorías, como personas, animales, objetos u otros tipos de cosas. El etiquetado de imágenes es un paso importante en el reconocimiento de imágenes, ya que permite a los modelos de aprendizaje automático aprender a identificar los objetos en las imágenes (Zeiler & Fergus, 2013).

**Entrenamiento de modelos:** El entrenamiento de modelos es el proceso de enseñar a los modelos de aprendizaje automático a identificar los objetos en las imágenes. El proceso de entrenamiento de modelos se lleva a cabo utilizando un conjunto de datos de imágenes que han sido etiquetadas. El modelo aprende a identificar los objetos en las imágenes comparando las imágenes etiquetadas con las imágenes sin etiquetar.

**Evaluación de modelos:** La evaluación de modelos es el proceso de medir el rendimiento de los modelos de aprendizaje automático. La evaluación de modelos se realiza utilizando un conjunto de datos de imágenes no etiquetadas. El modelo se utiliza para predecir las etiquetas de las imágenes no etiquetadas. El rendimiento del modelo se mide comparando las etiquetas predichas con las etiquetas reales (Acuña D. , 2019).

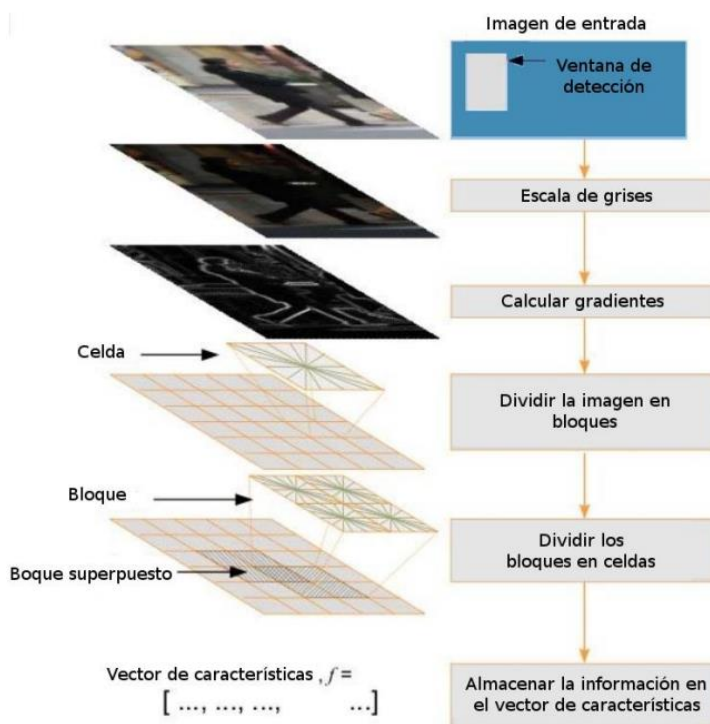
En este último punto se dará a conocer algunos modelos y sus respectivas características las cuales servirán para el desarrollo del presente proyecto de grado:

### Métodos de extracción de características

En el reconocimiento de imágenes, los métodos de extracción de características juegan un papel crucial al permitir que los algoritmos de aprendizaje automático comprendan y representen las imágenes de manera efectiva. Estos métodos buscan identificar y seleccionar las características más relevantes y distintivas presentes en las imágenes, lo que facilita la tarea de clasificación o detección (Jain & Vailaya , 1999), este ejemplo es más visible en la Figura 23.

### Figura 20

*Proceso de extracción de características de una imagen*



*Nota.* En la figura se puede apreciar es el proceso de extracción de características en una imagen. Tomado de trabajo de fin de grado |*Generación de algoritmos para clasificación de obstáculos*, (Ramos Alonso, 2016)

Algunos de los métodos de extracción de características más utilizados son:

*Extracción de características manuales:* Este enfoque implica la selección y diseño manual de características visuales, como texturas, bordes, formas o colores, para representar las imágenes (Fleet & Viola , 2001). Aunque fue ampliamente utilizado en el pasado, este método puede ser costoso y subjetivo, ya que depende del conocimiento y experiencia de los expertos (En la presente tesis se cuenta con el soporte de la Dra. Vanessa Abad, Ph.D. en Biotecnología).

*Descriptores de características locales:* Estos métodos buscan identificar puntos clave o regiones de interés en las imágenes y describir las características locales en torno a estos puntos. Ejemplos populares de descriptores locales incluyen SIFT (Scale-Invariant Feature Transform) y SURF (Speeded-Up Robust Features). Estos descriptores son invariantes a transformaciones como rotación y cambio de escala, lo que los hace efectivos en tareas de reconocimiento de objetos.

*Redes Neuronales Convolucionales (CNN):* En la última década, las redes neuronales convolucionales han adquirido un dominio significativo en el campo del reconocimiento de imágenes, gracias a su habilidad para aprender automáticamente características jerárquicas y abstractas directamente de los datos. En lugar de depender de características manuales o descriptores locales, las CNN utilizan filtros convolucionales para extraer automáticamente características relevantes de diferentes niveles de abstracción, lo que ha llevado a un gran avance en la precisión del reconocimiento de imágenes (Huang , Bottou , & LeCun, 2004).

*Aprendizaje por transferencia:* En lugar de diseñar y entrenar desde cero una CNN específica para una tarea de reconocimiento, se puede utilizar un modelo preentrenado en un gran conjunto de datos, como ImageNet. Luego, se pueden transferir las capacidades de extracción de características del modelo preentrenado a una tarea específica de reconocimiento de imágenes con un conjunto de datos más pequeño.

Estos métodos de extracción de características son fundamentales en el reconocimiento de imágenes, ya que permiten transformar la información visual en representaciones más significativas y compactas, lo que facilita la tarea de clasificación, detección o cualquier otra tarea que implique procesar imágenes. El aprendizaje profundo, en particular, ha demostrado ser especialmente efectivo en la extracción automática de características, lo que ha llevado a mejoras significativas en diversas aplicaciones del reconocimiento de imágenes. A continuación, se detalla algunos ejemplos de algunos métodos de extracción de características:

### **Extractor de características HAAR**

El método de extracción de Haar es un método de extracción de características para el reconocimiento de objetos. Fue propuesto por Viola y Jones en su artículo "Rapid object detection using boosted cascade of simple features" (Viola & Jones, 2001). El método de extracción de Haar funciona calculando la suma de los valores de píxeles en un conjunto de regiones de interés (ROI) de una imagen. Las ROI se seleccionan para que sean características de los objetos que se desea detectar. Por ejemplo, una ROI puede ser una región de la cara que contiene los ojos, la nariz y la boca.

Esta técnica se basa en el uso de características rectangulares simples llamadas "Haar-like" que capturan cambios en la intensidad de los píxeles en diferentes regiones de la imagen.

Para aplicar el método Haar, se utilizan ventanas rectangulares de diferentes tamaños y formas que se deslizan por toda la imagen. En cada posición de la ventana, se calcula la diferencia de suma de los píxeles en las regiones adyacentes, lo que genera características Haar para esa ubicación. Las características Haar pueden ser de diferentes tipos, como bordes, líneas o esquinas, y son sensibles a cambios abruptos de intensidad en la imagen. Estas características son útiles para identificar patrones específicos en objetos, como los bordes de un ojo o la línea de la boca en un rostro.

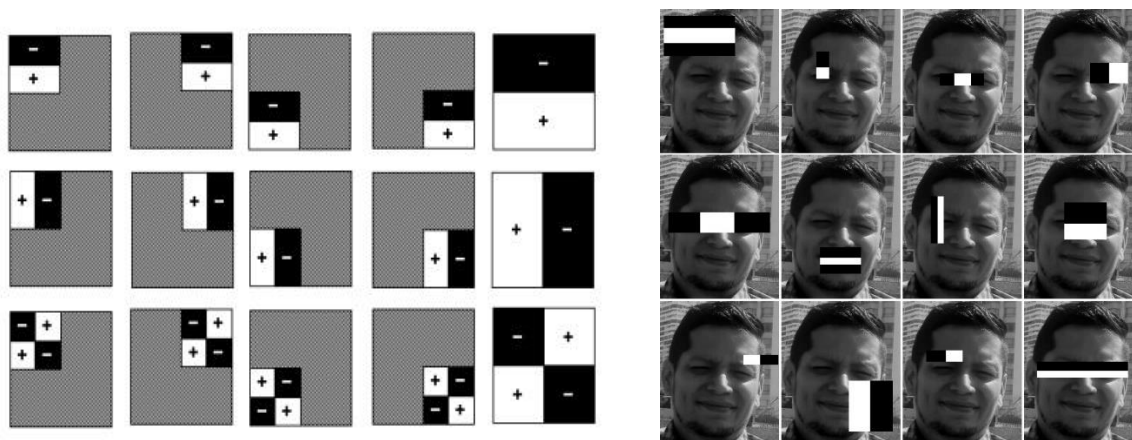


Una cascada de Haar es una estructura jerárquica que consiste en múltiples etapas o capas de clasificadores. Cada etapa contiene varios clasificadores que se combinan para formar una cascada, donde los clasificadores en etapas posteriores se activan solo si los de etapas anteriores han pasado la detección. Esto permite reducir significativamente el número de regiones de interés a examinar en la imagen, aumentando la eficiencia del proceso de detección (Mordvintsev & Abid , 2013).

El proceso de entrenar una cascada de Haar implica utilizar un conjunto de imágenes positivas que contienen el objeto de interés (por ejemplo, rostros) y un conjunto de imágenes negativas que no lo contienen. Luego, se extraen las características Haar de estas imágenes y se utilizan para entrenar el clasificador mediante el algoritmo AdaBoost. Una vez entrenada la cascada de Haar, se puede utilizar para detectar el objeto de interés en nuevas imágenes. La cascada de Haar examinará rápidamente las diferentes regiones de la imagen y detectará el objeto si coincide con las características aprendidas durante el entrenamiento como se aprecia en la Figura 21.

**Figura 21**

*Izq: Filtros Haar, Der: Filtros Haar escalados en diferentes posiciones de la imagen.*



*Nota.* En la figura se puede apreciar como los filtros HAAR van analizando la imagen y extrayendo la información. Tomado del sitioweb [carlosjuliopardoblog.wordpress.com](http://carlosjuliopardoblog.wordpress.com) | *Detección de Rostros basado en Filtros Haar + Adaboost*, (Pardo C. , 2023)

En resumen, Haar Cascade es una técnica de detección de objetos basada en el uso de características Haar y una estructura jerárquica de clasificadores, lo que permite una detección rápida y eficiente de objetos en imágenes. Es ampliamente utilizada en aplicaciones de detección de rostros, detección de ojos, reconocimiento de gestos y otras tareas de visión por computadora (Lienhart, Maydt, Huiskes, & Cremers, 2002).

### **HOG (Histograma De Gradientes Orientados)**

El método HOG, o Histograma de Gradientes Orientados, es una técnica ampliamente utilizada en visión por computadora para analizar imágenes y detectar objetos. Sefundamentación se basa en la idea de que las formas y contornos de los objetos se pueden identificar a través de la distribución de los gradientes de intensidad en la imagen. Para aplicar el método HOG, primero se divide la imagen en pequeñas celdas. Luego, se calculan los gradientes de intensidad dentro de cada celda en diferentes direcciones. Estos gradientes representan las transiciones de intensidad en la imagen y ayudan a capturar las características locales de los objetos (Dalal & Triggs, 2005).

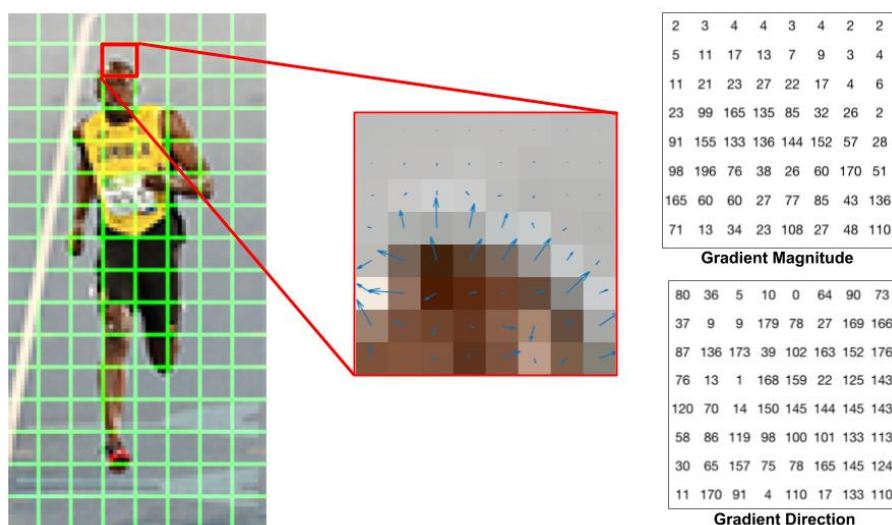
A continuación, los gradientes de cada celda se agrupan en histogramas que cuentan la cantidad de gradientes con ciertas orientaciones. Esto permite resumir la información sobre la dirección de los bordes presentes en cada celda.

Luego, los histogramas de todas las celdas se combinan para formar un vector de características que describe la apariencia general de la imagen en términos de sus gradientes direccionales.

El vector de características resultante puede ser utilizado para entrenar un clasificador que detecte objetos específicos en nuevas imágenes tal como se muestra en la Figura 22. La ventaja del método HOG radica en su capacidad para capturar patrones de bordes y formas en una imagen, lo que lo hace especialmente efectivo en tareas de detección de objetos (Zhang & Zhang).

## Figura 22

*Proceso de extracción de características de por HOG*



*Nota.* En la figura se puede apreciar como los filtros HOG van analizando la imagen y extrayendo la información. Tomado del sitio web LearnOpenCV.com | *Histogram of Oriented Gradients explained using OpenCV*, (LearnOpenCV, 2016)

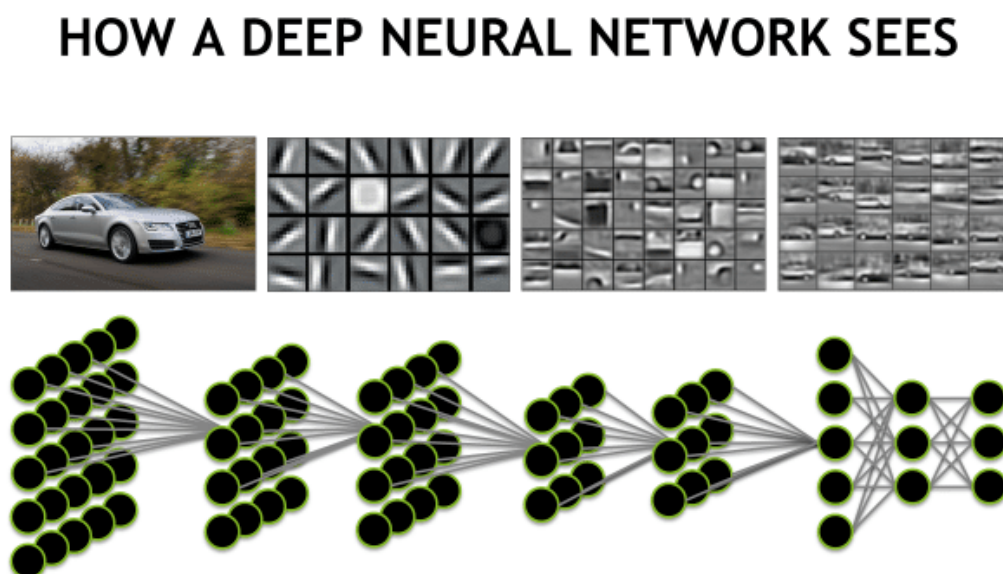
El método HOG ha sido ampliamente aplicado en diversos campos, como la detección de peatones en imágenes de vigilancia, reconocimiento de rostros en fotografías y detección de objetos en general. Su enfoque en el análisis de gradientes de intensidad y la generación de vectores de características lo convierten en una herramienta valiosa para resolver problemas de reconocimiento de objetos en el campo de la visión por computadora (Wang & Han, 2014).

## Las redes neuronales convolucionales (CNN)

Las redes neuronales convolucionales (CNN) son un subtipo de redes neuronales artificiales (ANN) ampliamente empleadas en aplicaciones de visión artificial. Estas redes son altamente efectivas para aprender características específicas de las imágenes, lo que las convierte en herramientas útiles para diversas tareas como clasificación de imágenes, detección de objetos y segmentación de imágenes.

### Figura 23

*Extracción de características por redes neuronales convolucionales*



*Nota.* En la figura se puede apreciar como la red neuronal convolucional extrae información de las imágenes. Tomado del sitioweb [nvidia.com](https://www.nvidia.com) | *Visión de Computación*, (Nvidia, 2023)

Las CNN están formadas por múltiples capas de filtros convolucionales. Los filtros convolucionales son funciones matemáticas que se utilizan para detectar patrones en las imágenes. Los filtros se emplean en una secuencia de capas para procesar las imágenes, donde cada capa aprende un conjunto nuevo de características, como se ilustra en la Figura 23.

Las redes neuronales convolucionales (CNN) son altamente eficaces para aprender características de las imágenes debido a su enfoque en el proceso de aprendizaje supervisado. En el aprendizaje supervisado, las CNN se entrenan con un conjunto de datos de imágenes que ya están etiquetadas con las características que se desea que aprendan. Por ejemplo, una CNN podría entrenarse con un conjunto de datos de imágenes de gatos y perros. Las imágenes de cada categoría estarían etiquetadas con la categoría correspondiente. Las CNN aprenden las características de las imágenes mediante un proceso llamado propagación hacia atrás. En la propagación hacia atrás, las CNN calculan el error entre las predicciones que realizan y las etiquetas reales. El error se utiliza para ajustar los pesos de los filtros, lo que permite a las CNN aprender nuevas características (Simonyan & Zisserman, 2014).

Dentro de las redes neuronales convolucionales pueden tener diversos filtros muy complejos para aplicaciones más complejas como clasificar objetivos personalizados, características especiales de una imagen, etc. A menudo estos filtros se encuentran en las últimas capas de la red.

Los patrones y características que generalmente busca son:

- Objetos
- Texturas
- Curvas
- Colores
- Bordes
- formas

Las redes neuronales convolucionales (CNN) se inician con una fase de extracción de características. Esta fase utiliza neuronas convolucionales para detectar patrones en las imágenes. Las neuronas convolucionales se aplican a las imágenes en una serie de capas, cada una de las cuales aprende un nuevo conjunto de características.

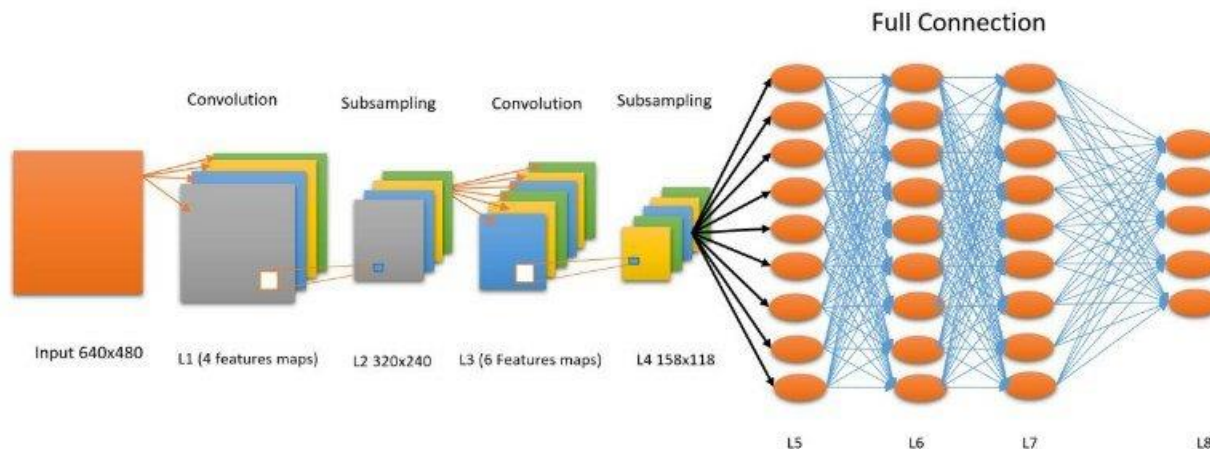
Después de la fase de extracción de características, se realiza una disminución por muestreo. La disminución por muestreo reduce el tamaño de la imagen, lo que hace que sea más fácil para las neuronas de perceptrón realizar la clasificación final. Las neuronas de perceptrón son neuronas que clasifican las imágenes basándose en las características que han sido extraídas por las neuronas convolucionales (He, Zhang, Ren, & Sun, 2016).

Un modelo de red neuronal bien entrenado debe tener un sesgo bajo y una varianza baja. El sesgo es la tendencia del modelo a cometer errores en los datos de entrenamiento. La varianza es la tendencia del modelo a cometer errores en los datos de prueba. Un modelo con un sesgo bajo es capaz de aprender las características de los datos de entrenamiento, pero puede no ser capaz de generalizar a nuevos datos. Un modelo con una varianza baja es capaz de generalizar a nuevos datos, pero puede no ser capaz de aprender las características de los datos de entrenamiento.

Una forma de reducir el sesgo es aumentar el tamaño del conjunto de datos de entrenamiento. Esto permite al modelo aprender más características de los datos. Otra forma de reducir el sesgo es utilizar una red neuronal más compleja. Esto permite al modelo aprender relaciones más complejas entre los datos. Una forma de reducir la varianza es utilizar una red neuronal más simple. Esto hace que el modelo sea menos propenso a aprender características aleatorias de los datos. Otra forma de reducir la varianza es regularizar la red neuronal. Esto significa añadir restricciones a los pesos de la red neuronal, lo que hace que sea menos probable que la red neuronal aprenda características aleatorias.

## **Figura 24**

*Reconfiguración de red neuronal convolucional*



*Nota.* En la figura se presenta la estructura fundamental de una red neuronal convolucional, que está conectada a una capa de clasificación. Tomado de *Aplicación de Deep Learning en Robótica Móvil para Exploración y Reconocimiento de Objetos basados en Imágenes*, (De la Rosa, 2016)

La Figura 24 ilustra el diseño elemental de una red neuronal convolucional. Una red neuronal convolucional está formada por capas de neuronas convolucionales y capas de neuronas de perceptrón. Las neuronas convolucionales se utilizan para detectar patrones en las imágenes. Las neuronas de perceptrón se utilizan para clasificar las imágenes basándose en los patrones que han sido detectados por las neuronas convolucionales.

- **Capa de entrada:** La imagen está compuesta por píxeles, los cuales tienen tres dimensiones: altura, anchura y profundidad. Estas dimensiones pueden consistir en 1 o 3 canales, que representan la escala de color de la imagen. En ocasiones puede tener y poseer ángulos de rotación del objetivo o característica a analizar de la imagen.
- **Capa De Convolución:** Analiza la imagen con una ventana deslizante. La ventana deslizante se mueve por la imagen, y cada vez que se detiene en una posición, calcula el producto escalar entre los valores de la ventana y los pesos de la capa convolucional. El producto escalar es una medida de la similitud entre la ventana y los pesos de la capa

convolucional. La capa convolucional produce una matriz de números, que se conoce como mapa de características. El mapa de características representa las características más importantes de la imagen. Para extraer y resaltar diferentes patrones y características en el ejemplo, se emplean una serie de filtros en la figura se utiliza 32 filtros, se puede decidir la cantidad de filtros que se necesite dependiendo la aplicación siendo este aspecto el volumen de salida

- **Capa ReLu:** Permite obtener los datos positivos y dejar de lado los datos negativos por medio de una función de activación en cada elemento de la matriz, siendo esta la que permite a la red aprender características más complejas.
- **Capa pooling:** Reduce el tamaño del mapa de características, lo que hace que sea más fácil para la red clasificar la imagen.
- **Capa de Clasificación:** La capa de clasificación tiene una neurona por categoría. La neurona que tenga la salida más alta es la categoría a la que se clasifica la imagen.

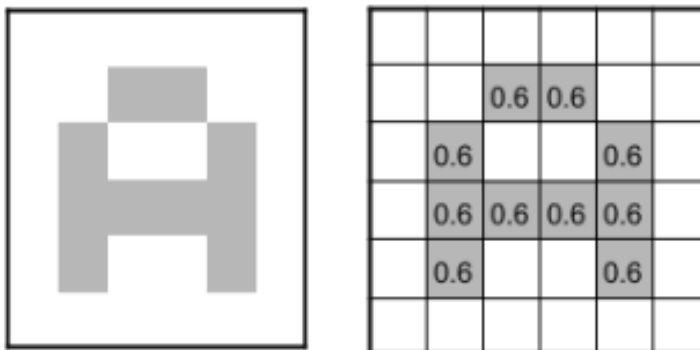
### **Aprendizaje de una red neuronal convolucional**

Una red neuronal convolucional es una serie de filtros los cuales obtienen patrones que permites obtener información que ayudar a entrenar el modelo, como ejemplo se tiene una imagen con dimensiones de 28x28, esta imagen ingresa al sistema mediante su digitalización, de esta forma la maquina en este caso el PC le permita entender esos datos, tal como muestra en la Figura 25.



**Figura 25**

Lado izquierdo: Fotografía en tonos de gris para ser procesada; Lado derecho: Fotografía transformada en una matriz de píxeles.

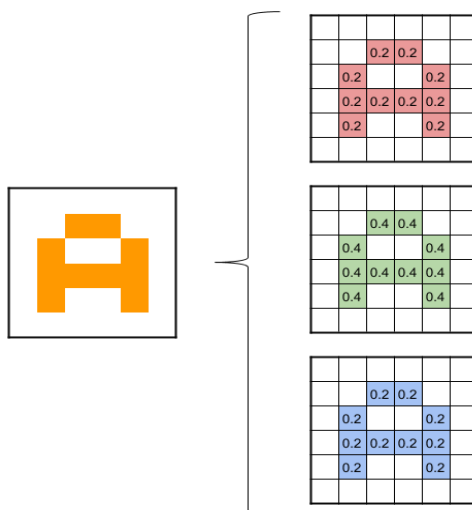


En la ilustración, se puede notar que en el lado izquierdo se presenta la imagen física que será digitalizada. En cambio, en la figura de la derecha se puede apreciar el procesamiento de la imagen, donde se representa mediante una matriz de píxeles con valores que varían de 0 a 255. Luego, a través de un proceso de normalización, los datos se escalan para quedar dentro del rango de 0 a 1. Tomado de *Redes Neuronales Convolucionales*, (Barrios, 2022)

A continuación, la matriz resultante de la imagen procesada se utiliza como capa de entrada para la red. En este ejemplo, se emplea una imagen con dimensiones de 28x28, lo que se traduce en un total de aproximadamente 784 neuronas en este momento. Cabe señalar que la imagen está en escala de grises, pero dado que las imágenes estándar se componen de tres colores (escala RGB), se multiplica por 3, lo que resulta en un total de 2352 neuronas para la capa de entrada de la red neuronal convolucional, tal como se observa en la Figura 26.

## Figura 26

Lado izquierdo: Imagen a color a ser procesada; Lado derecho: Imagen transformada en tres matrices de píxeles correspondientes a su escala en RGB.



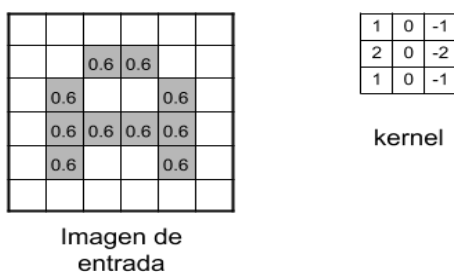
*Nota.* La imagen presenta el procesamiento en la escala de píxeles ya normalizada, dividida en matrices. Tomado de *Redes Neuronales Convolucionales*, (Barrios, 2022)

Antes de ingresar los datos a la red, es preferible realiza un proceso de normalización, esto se debe a que los datos de imagen suelen estar en un rango de 0 a 255, lo que puede ser demasiado grande para la red neuronal. La normalización divide cada valor de datos por 255, lo que lo escala a un rango de 0 a 1, se puede observar la Figura 25 y Figura 26.

A continuación, se inicia las convoluciones, ya que además es el paso más importante en una red neuronal convolucional. En este paso, la red neuronal toma un conjunto de píxeles adyacentes de la imagen de entrada y los utiliza para calcular un valor de salida. El tamaño del conjunto de píxeles que se utiliza se llama kernel. El kernel suele tener un tamaño de 3x3 píxeles, pero puede tener otro tamaño según la aplicación. La red neuronal utiliza el kernel para calcular una matriz de salida, que se llama capa de neuronas ocultas, ver Figura 27.

**Figura 27**

Izquierda: Imagen de a procesar; Derecha: aplicación del Kernel o filtro.

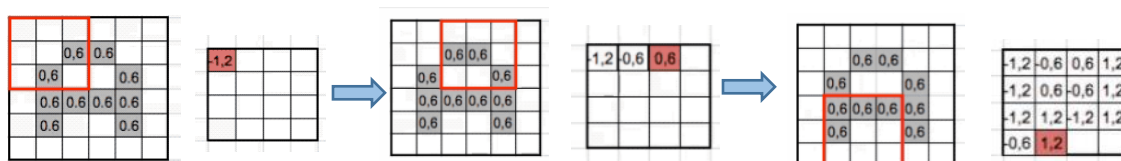


*Nota.* La imagen a la izquierda muestra el kernel, el cual se utiliza para realizar un proceso de recorrido dentro de la imagen, obteniendo las características necesarias para llevar a cabo la convolución.. Tomado de *Redes Neuronales Convolucionales*, (Barrios, 2022)

Una de las características de las redes neuronales convolucionales que las hace tan eficaces es que pueden utilizar múltiples kernels para procesar una imagen. En la Figura 28, se utilizan 32 kernels para procesar la imagen de 224x224 píxeles. Esto resulta en una capa oculta de 25088 neuronas. Si la imagen fuera a color, habría 3 canales (rojo, verde y azul), por lo que habría 3 capas ocultas, cada una con 25088 neuronas.

**Figura 28**

*Recorrido del kernel en la imagen extrayendo características*

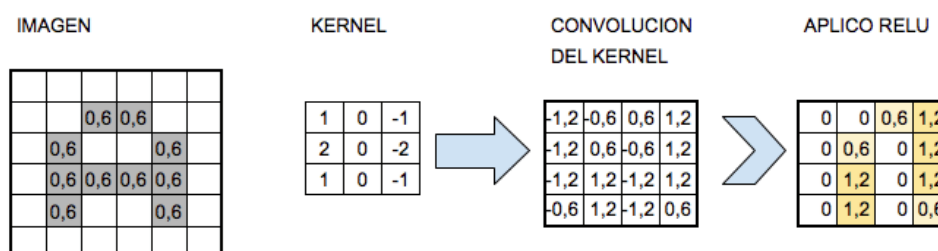


*Nota.* En la figura se puede apreciar cómo el kernel se desplaza a lo largo de la imagen, capturando características relevantes para el proceso de convolución. Tomado de *Redes Neuronales Convolucionales*, (Barrios, 2021).

Conforme el kernel se mueve a través de la imagen, se genera una nueva imagen que ha sido filtrada por el kernel. Esto se puede ver en la Figura 28. En la primera convolución, se crean 32 imágenes filtradas nuevas. Estas nuevas imágenes capturan ciertos rasgos y patrones de la imagen original. Estos rasgos y patrones se pueden utilizar para diferenciar entre diferentes objetos en la imagen, como un animal, automóvil o un objeto personalizado a reconocer.

## Figura 29

### Implementación de la función ReLu



*Nota.* En la imagen se ilustra cómo la función ReLu se aplica después de convolucionar con el kernel, separando los valores positivos de los negativos. Tomado de *Redes Neuronales Convolucionales*, (Murphy, 2012)

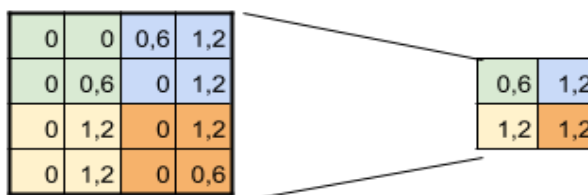
Una vez que la imagen ha sido filtrada por el kernel, se utiliza una función de activación para transformar los valores. Una de las funciones de activación más comunes se llama ReLU, que significa unidad lineal rectificadora. La función ReLU transforma los valores negativos a 0 y los valores positivos a sí mismos. Esto ayuda a la red neuronal a aprender características que son importantes para la tarea en cuestión. Por ejemplo, si la red neuronal está tratando de identificar coches en imágenes, la función ReLU puede ayudar a la red a aprender que las esquinas y los bordes las cuales son características importantes de los carros. La función ReLU también puede ayudar a la red neuronal a aprender a ignorar características que no son importantes para la tarea, como el fondo de la imagen ver Figura 29.

## Proceso de muestreo

Después de la primera capa convolucional, esta tiene una capa oculta con 25.088 neuronas. Si se realizara otra capa convolucional, la cantidad de neuronas en la siguiente capa sería significativamente mayor. Para reducir la cantidad de neuronas, la red neuronal utiliza un proceso llamado muestreo. El muestreo consiste en seleccionar una muestra de las neuronas de la capa oculta. Las neuronas seleccionadas son las que tienen los valores más altos. El muestreo ayuda a reducir la cantidad de neuronas en la red neuronal, lo que reduce el poder computacional necesario para entrenar la red, tal como se observa en la Figura 30. También ayuda a mejorar la precisión de la red, ya que la red se centra en las neuronas más importantes.

### Figura 30

*Implementación del muestreo o subsampling y Max-Pooling*



*Nota.* La figura exhibe la aplicación de la técnica denominada muestreo o subsampling, específicamente Max-Pooling, que selecciona los valores más relevantes y reduce la salida a la mitad. Tomado de *Redes Neuronales Convolucionales*, (Murphy, 2012)

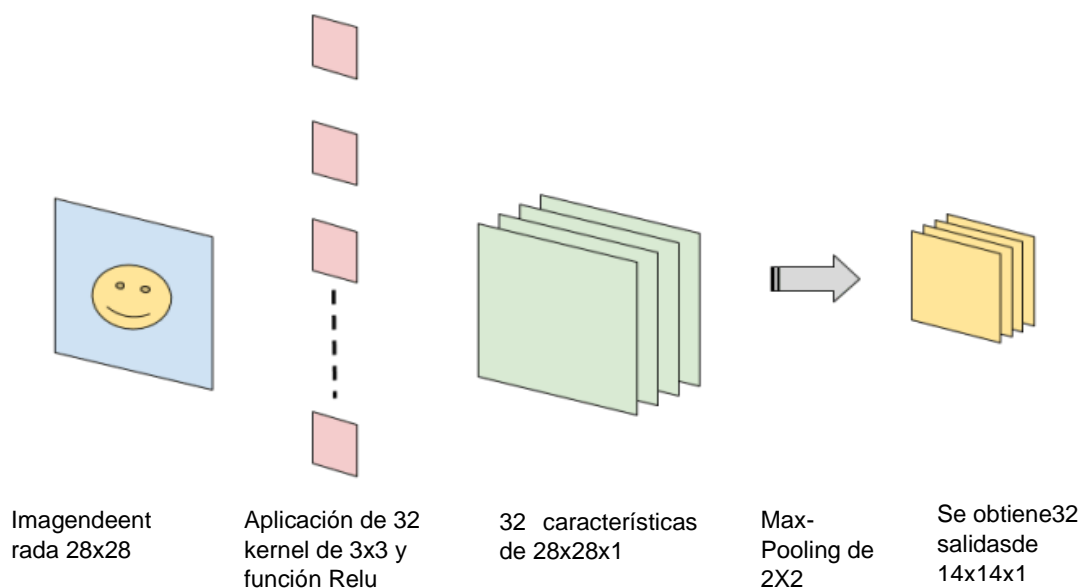
El muestreo es un proceso que se utiliza para reducir la cantidad de neuronas en una red neuronal convolucional. El muestreo más utilizado se llama Max-Pooling. El Max-Pooling funciona tomando un bloque de 2x2 píxeles y conservando el valor más alto de cada bloque. Esto reduce la cantidad de neuronas en la red neuronal a la mitad.

El Max-Pooling tiene varias ventajas. En primer lugar, ayuda a reducir el poder computacional necesario para entrenar la red neuronal. En segundo lugar, ayuda a mejorar la precisión de la red neuronal, ya que la red se centra en los píxeles más importantes de la imagen.

En el ejemplo que proporcionó, la imagen original tenía un tamaño de 28x28 píxeles. Después de una capa de Max-Pooling, la imagen tenía un tamaño de 14x14 píxeles. Esto representa una reducción de la cantidad de neuronas en la red neuronal de 25.088 a 6272. El Max-Pooling también puede ayudar a mejorar la velocidad de la red neuronal. Esto se debe a que el Max-Pooling reduce la cantidad de cálculos que necesita realizar la red neuronal, tal como se muestra en la Figura 31.

**Figura 31**

*Primera convolución*



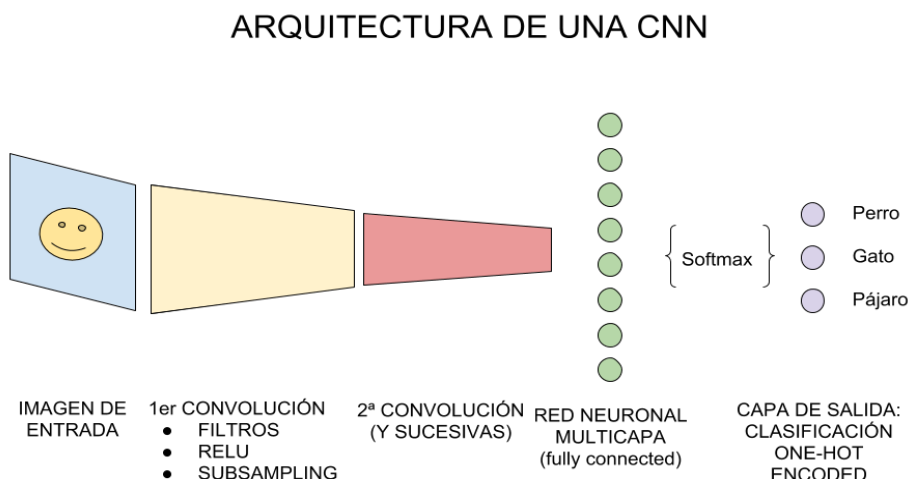
*Nota.* La figura proporciona una visión resumida de cómo se ejecutan los procesos de convolución para obtener las salidas con las características de las imágenes que se están procesando. Tomado de *Introducción a la Visión Artificial-Visión Artificial Avanzada*, (Fernández Castilla, 2013)

A medida que se agregan más capas convolucionales, los mapas de características pueden identificar formas más complejas. Esto se debe a que cada capa convolucional aprende un conjunto diferente de características, y las capas convolucionales posteriores pueden utilizar las características aprendidas por las capas convolucionales anteriores para aprender características más complejas. El conjunto total de capas convolucionales puede identificar el objeto que se está entrenando el sistema. Esto se debe a que las capas convolucionales pueden aprender un conjunto de características que son específicas del objeto. La última capa oculta se aplana. Esto significa que la última capa oculta se convierte en una capa de neuronas tradicionales. Este procedimiento se lleva a cabo con el fin de permitir que la red neuronal utilice la función softmax para clasificar las entradas en diferentes categorías.

La última capa oculta tal como se observa en la Figura 32, se conecta a una nueva capa oculta de neuronas tipo feedforward. Estas neuronas tipo feedforward utilizan la función softmax para clasificar las entradas en categorías.

### Figura 32

*Integración a una red convencional*



*Nota.* La figura proporciona una representación visual de cómo están compuestas las redes neuronales convolucionales, incluyendo sus diversos mecanismos y componentes. Tomado de *Introducción a la Visión Artificial-Visión Artificial Avanzada*, (Aprende Machine learning, 2021)

La función softmax es una función matemática que se utiliza para convertir las salidas de una red neuronal en probabilidades. La función softmax asigna una probabilidad a cada clase, y la suma de todas las probabilidades es 1. Esto permite a la red neuronal clasificar las entradas en categorías. En el ejemplo que proporcionó, la red neuronal está clasificando perros y gatos. En consecuencia, la capa de salida consta de dos neuronas, una asignada a la categoría de perros y otra a la de gatos. Del mismo modo, si se clasifican objetos como zapatos, pelotas o raquetas, habrá tres neuronas en la capa de salida, y así sucesivamente, dependiendo del número de clases que pueda reconocer el sistema. Esto permitirá a la red neuronal clasificar las imágenes de perros y gatos con precisión.

### **Otras arquitecturas para extracción de características.**

*LBP (Local Binary Patterns):* Los Patrones Binarios Locales son una técnica utilizada para describir la textura de una imagen. Se basan en comparar el valor de intensidad de un píxel central con sus vecinos y codificarlos como números binarios. Los histogramas de los patrones binarios locales se utilizan como características para representar la textura de la imagen (Bay, Ess, Tuytelaars, & Van Gool, 2006).

*SIFT (Scale-Invariant Feature Transform):* El SIFT es una técnica de detección y descripción de características invariante a la escala y la rotación. Extrae puntos clave o características distintivas de la imagen que son resistentes a las transformaciones geométricas, lo que los hace útiles para tareas como el reconocimiento de objetos y el emparejamiento de imágenes (Felzenszwalb, Huttenlocher, & Belongie, 2004).



*SURF (Speeded-Up Robust Features)*: Similar al SIFT, el SURF es otro método para detección y descripción de características que es más rápido y eficiente computacionalmente. También es invariante a la escala y la rotación y se utiliza en aplicaciones como la detección de objetos y el seguimiento de características en video (Mikolajczyk & Schmid, 2005).

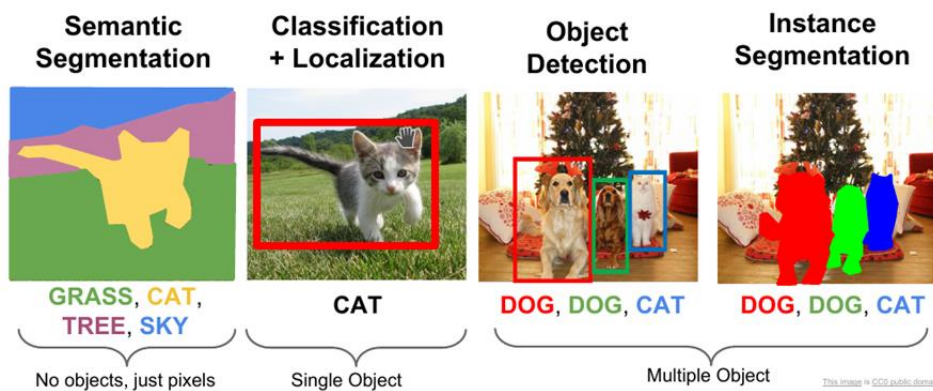
*Gabor Filters*: Los filtros de Gabor son filtros lineales utilizados para detectar patrones en diferentes direcciones y escalas. Son ampliamente utilizados en tareas de detección de texturas y reconocimiento de objetos en imágenes (Bell & Sejnowski, 1997).

### Métodos de clasificación

Dentro de los métodos de clasificación usando IA con el tema asociado a clasificación de imágenes, ver ejemplos en la Figura 33.

**Figura 33**

*Deep Learning en Visión Artificial*



*Nota.* La figura muestra las distintas aplicaciones de clasificación de imágenes aplicando visión por computadora. Tomado del sitio web: [blog.cartif.es](http://blog.cartif.es) | *Deep Learning en Visión Artificial*, (Duque Domingo, 2022)

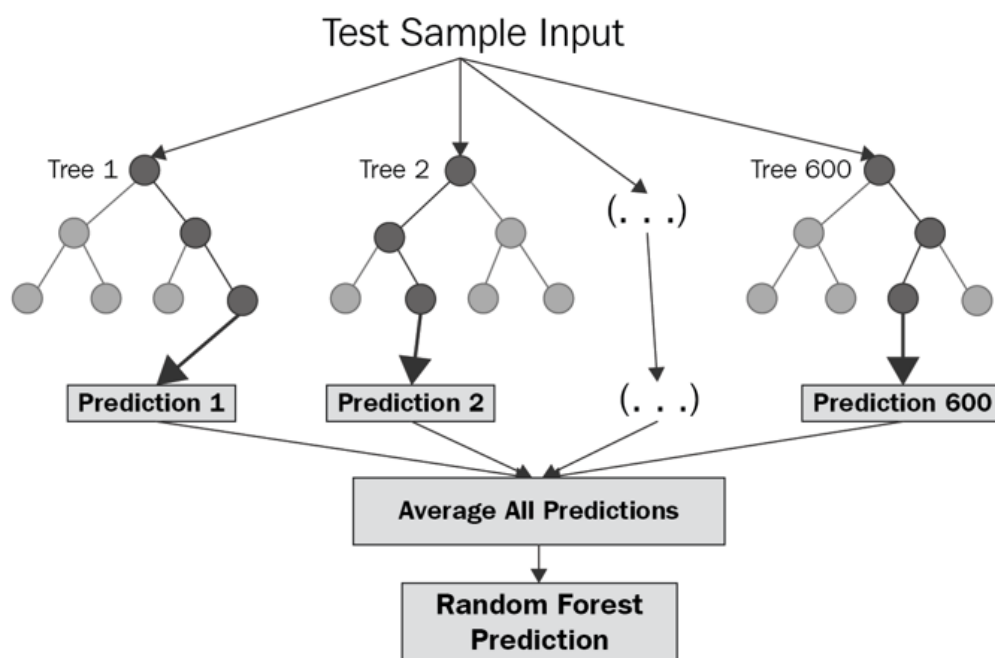
Algunos conceptos tenemos los siguientes conceptos que se explica a continuación:

## Bosques aleatorios (Random Forest)

Los bosques aleatorios son una categoría de algoritmos de aprendizaje automático supervisado que se utiliza para la clasificación y regresión. Los bosques aleatorios se componen de un conjunto de árboles de decisión independientes que se entrenan en diferentes subconjuntos del conjunto de datos (Breiman, 2001). Cada árbol de decisión vota por la clase o valor más probable, y el bosque aleatorio toma la decisión de la mayoría de los votos, ver en la Figura 34.

**Figura 34**

*Estructura de la red de bosques aleatorios formados por distintos arboles*



*Nota.* Se puede apreciar en la figura, el modelo de bosque aleatorio que no es más que diferentes árboles de decisión que su objetivo es mejorar la precisión y reducir el sobre ajuste. Tomado del sitio web: [siliconvalley.basisindependent.com](http://siliconvalley.basisindependent.com) | *Creación Del Modelo De Bosque Aleatorio Inicial*, (siliconvalley, 2021)

Los bosques aleatorios son un método muy poderoso para la clasificación de imágenes. Son capaces de aprender relaciones complejas entre los píxeles de una imagen y las clases de las imágenes. Los bosques aleatorios también son muy robustos a los datos ruidosos y a los datos mal etiquetados.

Existen algunos ejemplos de cómo se utilizan los bosques aleatorios para la clasificación de imágenes:

*Reconocimiento de objetos:* los bosques aleatorios se pueden utilizar para identificar objetos en imágenes. Por ejemplo, se pueden utilizar para identificar personas, vehículos o productos.

*Clasificación de imágenes médicas:* los bosques aleatorios se pueden utilizar para diagnosticar enfermedades en imágenes médicas. Por ejemplo, se pueden utilizar para diagnosticar cáncer de piel o enfermedades cardíacas.

*Clasificación de imágenes de satélite:* los bosques aleatorios se pueden utilizar para identificar diferentes tipos de terreno en imágenes de satélite. Por ejemplo, se pueden utilizar para identificar bosques, campos agrícolas analizando las plagas y enfermedades que puede atacar a una planta.

### **Maquinas soporte Vectorial (SVM)**

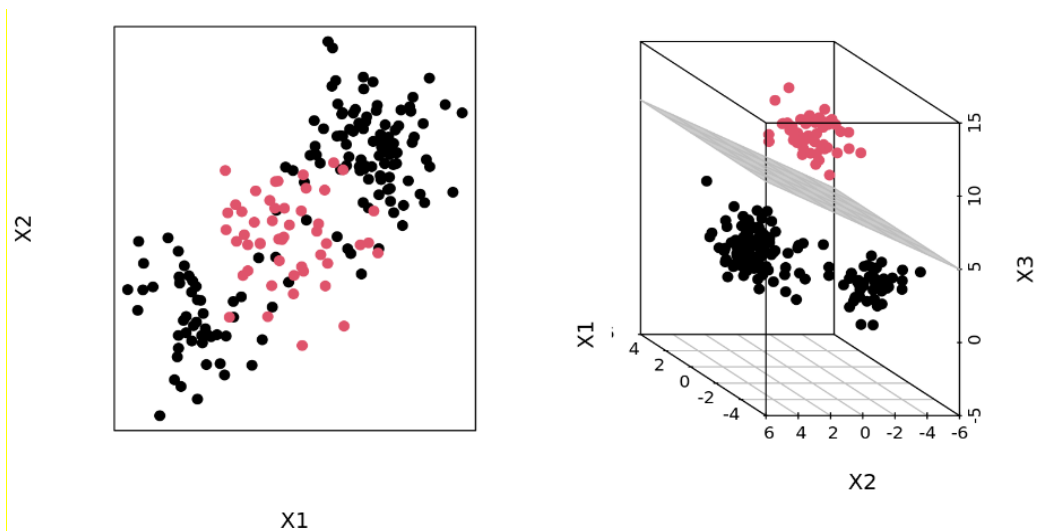
Las máquinas de soporte vectorial (SVM), también conocidas como Support Vector Machines en inglés, son empleadas tanto para tareas de clasificación como para regresión en predicciones, este método clasifica una imagen nueva identificando un hiperplano que separa las imágenes de diferentes clases (Cortes & Vapnik, 1995). Las imágenes que caen en un lado del hiperplano se clasifican como una clase, y las imágenes que caen en el otro lado del hiperplano se clasifican como otra clase

A continuación, se presentan algunos aspectos destacados de este modelo de aprendizaje automático:

- Las SVM se basan en la idea de encontrar un hiperplano que separe dos clases de datos de manera óptima.
- El hiperplano óptimo se refiere al plano que busca maximizar la distancia entre los puntos de soporte, que son aquellos puntos de datos que se encuentran más cercanos al hiperplano. Cuando los datos exhiben una estructura lineal, las Máquinas de Soporte Vectorial (SVM) pueden separarlos generando un margen amplio entre las dos clases. Esto significa que los puntos de datos de una clase estarán muy cerca del hiperplano, mientras que los puntos de datos de la otra clase estarán muy lejos del hiperplano este aspecto se puede observar en la Figura 35.

**Figura 35**

*Izquierda: datos lineales a clasificar, derecha: datos ya clasificados de forma lineal*



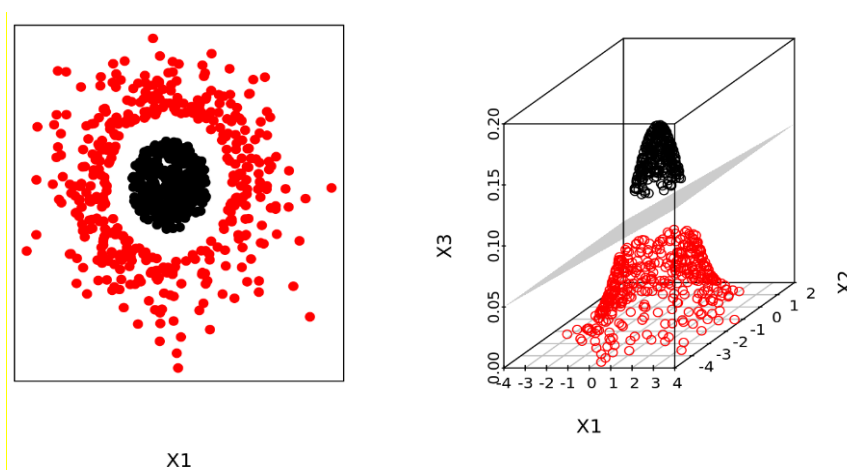
*Nota.* El gráfico ilustra el funcionamiento del modelo SVM en clasificación, el cual logra separar los datos lineales mediante una nueva dimensión. Tomado de sitio web: [cienciadedatos.net](http://cienciadedatos.net)

*[Máquinas de Vector Soporte (Support Vector Machines, SVMs), (Amat Rodrigo, 2017)*

En situaciones donde los datos no pueden ser separados de manera lineal, se recurre al uso de núcleos para transformarlos hacia un espacio de mayor dimensión donde sí pueden ser separados linealmente. Los núcleos se utilizan para transformar los datos de modo que se puedan separar con un hiperplano. Los tipos de núcleos más usuales son el núcleo lineal, el núcleo gaussiano y el núcleo polinomial (Hastie, Tibshirani, & Friedman, 2009).

### Figura 36

*Izquierda: datos no linealmente separables, derecha: datos ya clasificados no lineal*



*Nota.* El gráfico muestra en la parte izquierda se encuentran los datos no linealmente separables que deben ser clasificados, mientras que en la parte derecha se presentan los datos ya clasificados de forma no lineal. Tomado de sitio web: [cienciadedatos.net](http://cienciadedatos.net) | *Máquinas de Vector Soporte (Support Vector Machines, SVMs)*, (Amat Rodrigo, 2017)

El kernel lineal es el kernel más simple. Se calcula simplemente multiplicando las dos características de dos puntos de datos. El kernel gaussiano es más complejo. Se calcula multiplicando las dos características de dos puntos de datos y luego se aplica una función Gaussiana a la multiplicación. El kernel polinomial es aún más complejo. Se calcula multiplicando las dos características de dos puntos de datos y luego se aplica una función polinomial a la multiplicación. La Figura 36 muestra cómo los núcleos se pueden utilizar para mapear datos no linealmente separables a un espacio de alta dimensión donde los datos se vuelven linealmente

separables. En el gráfico de la izquierda, los datos no se pueden separar con un hiperplano porque no son linealmente separables. En el gráfico de la derecha, los datos se han mapeado a un espacio de alta dimensión utilizando un núcleo y ahora se pueden separar con un hiperplano.

Los núcleos son una herramienta poderosa que se puede utilizar para mejorar la precisión de las máquinas de soporte vectorial cuando los datos no son linealmente separables

### **Aprendizaje Profundo (Deep Learning DL)**

El aprendizaje profundo es una rama del aprendizaje automático que encuentra su inspiración en el funcionamiento del cerebro humano. Las redes neuronales profundas representan un tipo de algoritmo de aprendizaje profundo capaz de aprender de los datos y comprender el mundo mediante una jerarquía de conceptos. Estas redes están compuestas por múltiples capas de neuronas, donde cada una está conectada con las neuronas de las capas contiguas. Las neuronas en la primera capa reciben los datos de entrada, mientras que las de la última capa generan la salida. Las neuronas de las capas intermedias procesan los datos y aprenden los patrones entre los datos de entrada y la salida (Goodfellow, Bengio, & Courville, 2016).

Las redes neuronales profundas tienen la capacidad de ser entrenadas para llevar a cabo una amplia gama de tareas, que abarcan desde el reconocimiento de imágenes y el procesamiento del lenguaje natural hasta la traducción automática. Las redes neuronales profundas han sido muy exitosas en estas tareas, y han superado a otros algoritmos de aprendizaje automático en muchos casos. El crecimiento exponencial del aprendizaje profundo se debe a diversos factores, entre ellos, el incremento en la capacidad de procesamiento de las computadoras, el desarrollo de nuevos algoritmos de aprendizaje profundo y la disponibilidad de extensas bases de datos (LeCun, Bengio, & Hinton, 2015). El aumento de la potencia de las computadoras ha posibilitado el entrenamiento de redes neuronales más profundas y complejas..

El desarrollo de nuevos algoritmos de aprendizaje profundo ha hecho que las redes neuronales profundas sean más fáciles de entrenar y más precisas. La disponibilidad de grandes conjuntos de datos ha permitido entrenar redes neuronales deep con grandes cantidades de datos, lo que ha mejorado su rendimiento, siendo el aprendizaje profundo una herramienta poderosa que tiene el potencial de revolucionar una amplia gama de industrias. El aprendizaje profundo posee la capacidad de automatizar diversas tareas que actualmente son ejecutadas por seres humanos, además de abrir la puerta a nuevas aplicaciones que no pueden ser logradas mediante otros métodos de aprendizaje automático (Silver, y otros, 2016).

La figura 37 muestra el aumento de la potencia de las computadoras en los últimos años. El aumento de la potencia de las computadoras ha permitido entrenar redes neuronales más profundas y complejas, lo que ha mejorado el rendimiento del aprendizaje profundo.

### Figura 37

Concurso de clasificación de imágenes imageNET



*Nota.* La figura se puede observar las respuestas de los concursos de clasificación de imágenes hasta el 2017. Tomado de *ImageNet Large Scale Visual Recognition Challenge*, (Russakovsky, y otros, 2017)

De forma simplificada, el aprendizaje profundo es una herramienta tecnológica ampliamente utilizada en diversos campos industriales para enseñar a las computadoras a

aprender, modificar y adaptar, siguiendo el mismo proceso que los seres humanos han realizado desde tiempos remotos: aprender mediante el ejemplo, procesar y filtrar datos complejos utilizando los sentidos para obtener un resultado final y lograr mejoras continuas.

### **Otros métodos de clasificación**

Actualmente se ha tratado los algoritmos que mejor se adaptan a la solución del presente trabajo de titulación, no obstante, es importante conocer que existen diversos algoritmos de clasificación que se asocian a los diversos campos de la IA, los cuales se describen a continuación:

*Árboles de decisión:* Los árboles de decisión constituyen un tipo de algoritmo de aprendizaje automático que se fundamenta en la creación de un diagrama con forma de árbol para representar las decisiones que deben tomarse con el fin de llegar a una conclusión determinada. Los árboles de decisión se pueden utilizar para clasificar imágenes, identificando las características que son más importantes para la clasificación (Quinlan, 1986).

*Reglas de asociación:* Son un tipo de algoritmo de aprendizaje automático que se basa en la identificación de reglas que se asocian con frecuencia con ciertos eventos, esto viene de la mano con identificar las clases de las imágenes y así llegar a conocer a cuál pertenece (Aggarwal, 2014).

*Vecinos más cercanos:* Los vecinos más cercanos son un tipo de algoritmo de aprendizaje automático que se basa en la identificación de los vecinos más cercanos de un punto dado, así de esta forma se puede etiquetar a las imágenes clasificándolas (Cover & Hart, 1967).



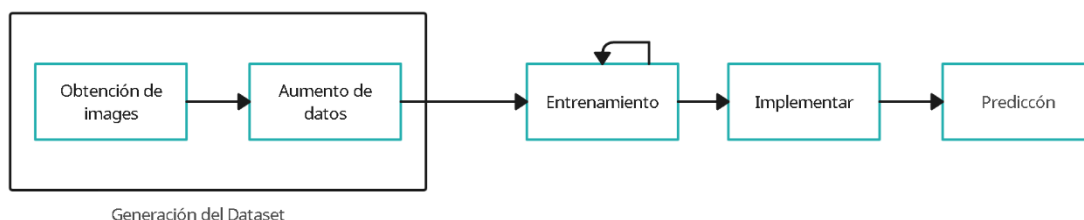
## Capítulo III

### Metodología y desarrollo de la solución

En este capítulo, se presenta la metodología utilizada para el desarrollo del algoritmo de solución en detección de la enfermedad “Tizón tardío” en el cultivo de papa. Se da a conocer los pasos realizados para la recolección y la preparación de datos, la creación y el entrenamiento del modelo. Además, se describen las herramientas, plataformas y tecnologías que se usaron durante el proceso. En la figura 38 se observa un diagrama de las actividades llevadas a cabo.

#### Figura 38

*Actividades realizadas en el proyecto*



*Nota.* Se muestran todas las actividades de inicio a fin ejecutadas para el funcionamiento del algoritmo de solución

#### Obtención de imágenes para la Base de Datos

La etapa inicial de la solución para el desarrollo del algoritmo de detección de enfermedades tizón tardío en el cultivo de papa, es la creación de una base de datos que contenga imágenes de hojas sanas y enfermas, para lo cual se toma un conjunto de datos de diferentes bases de datos abiertas y disponibles en los repositorios institucionales de Universidades y/o Grupos de Investigación, como lo son: Kaggle, PlantVillage (Universidad Estatal de Pensilvania) e ImageNet. A esta base de datos se la denominó “BaseData” y contiene

imágenes de hojas de planta de la papa, las cuales están divididas en tres categorías: Tizón tardío, tizón temprano y estado saludable.

### **PlantVillage**

El Proyecto PlantVillage es una base de datos de código abierto orientada hacia el diagnóstico de enfermedades de varios cultivos, desarrollado en la Universidad Estatal de Pensilvania en Estados Unidos, cuyo objetivo principal es ayudar a agricultores de todo el mundo con diagnósticos de enfermedades y dar soluciones mediante inteligencia artificial. En esta base de datos, se hallan imágenes de hojas de diversos cultivos, de las cuales se extrajeron imágenes de la papa con la enfermedad tizón tardío, tizón temprano y saludables.

### **Figura 39**

*Logo de PlantVillage*



*Nota.* La imagen muestra el logo de PlantVillage. Tomado de (plantvillage.psu.edu, 2013)

### **ImageNet**

Es un conjunto de datos y un sitio web que está centrado en recopilar y etiquetar imágenes para el uso en tareas de reconocimiento visual, visión por computadora y aprendizaje profundo. Representa uno de los repositorios más amplios y accesibles sin costo para estudios orientados a un recurso investigativo, en el ámbito del análisis de imágenes.

### **Figura 40**

*Logo Imagenet*



*Nota.* La imagen muestra el logo de IMAGENET. Tomado de (image-net.org, 2015)

## **Kaggle**

Es una plataforma en línea dedicada a proporcionar conjuntos de datos libres el uso con fines investigativos. El sitio web brinda a los usuarios acceder a una amplia variedad de datos, que van desde imágenes y texto hasta datos estructurados, y utilizarlos para entrenar y evaluar modelos de aprendizaje automático.

### **Figura 41**

*Logo de Kaggle*



*Nota.* La imagen muestra el logo de IMAGENET Kaggle. Tomado de (kaggle)

## **Consideraciones para la DataBase**

A la base de datos se le conformó con imágenes del tizón temprano ya que tanto el tizón temprano como el tizón tardío son enfermedades fúngicas similares en apariencia que afectan el cultivo de la papa.

A pesar de ser causadas por diferentes patógenos y tener algunas diferencias en su apariencia y comportamiento, también comparten parecidos, lo que puede hacer que sea difícil diferenciarlas a simple vista en algunas etapas de la infección por lo mencionado, se incluye a esta enfermedad en la Base de datos. Ambas enfermedades causan manchas en las hojas de papa. En el caso del tizón temprano, las manchas son de forma redonda u ovalada y pueden tener un centro más claro y un borde más oscuro. En el tizón tardío, las manchas son más grandes y generalmente tienen un borde irregular y de apariencia acuosa de color café, ver Figura 42. (Acuña & Gutiérrez, 2004)

**Figura 42**

*Hojas infectadas con el tizón tardío*



*Nota.* La imagen muestra hojas de papa con la enfermedad tizón tardío con manchas café y apariencia acuosa tomado de sitio web:[tizon.inia.c](http://tizon.inia.c)|*Cómo reconocer los tizones de* (Acuña & Gutiérrez, 2004)

**Figura 43**

*Hojas infectadas con el tizón temprano*



*Nota.* En la imagen se puede observar hojas de papa con la enfermedad tizón temprano con las zonas café son manchas circulares marrones con anillos concéntricos tomado del sitio web:[tizon.inia.c](http://tizon.inia.c)|*Cómo reconocer los tizones de* (Acuña & Gutiérrez, 2004)

Se compiló todas las imágenes obtenidas en una base de datos llamada “DataBase”

con la siguiente distribución:

**Tabla 1**

*Número de imágenes y clases*

| Clase | Estado de la planta | Número de imágenes |
|-------|---------------------|--------------------|
| 0     | Saludable           | 152                |
| 1     | Tizón Tardío        | 1000               |
| 2     | Tizón Temprano      | 1000               |

*Nota.* Esta tabla muestra el número total de muestras por cada clase para ser empleada en el algoritmo.

### **Preparación de datos**

Los datos usados para crear el modelo de solución deben estar adaptados para el entrenamiento, la validación y la prueba, por lo que a las imágenes se les debe realizar ciertos cambios como son: redimensionamiento, aumento de datos, normalización, etc.

### **Tamaño de las imágenes**

En aplicaciones de visión por computadora y detección, se utilizan imágenes de 256x256 píxeles es que proporcionan un correcto equilibrio entre la cantidad de detalles visuales que proporcionan, en este caso la enfermedad tizón tardío y tizón temprano, y la cantidad de recursos computacionales que requieren para procesar. Imágenes de una mayor resolución toman un tiempo considerablemente elevado y requieren más memoria para el procesamiento al trabajar con grandes conjuntos. A su vez, imágenes de menor resolución no proporcionan los suficientes detalles visuales para que el modelo de IA pueda realizar tareas con precisión.

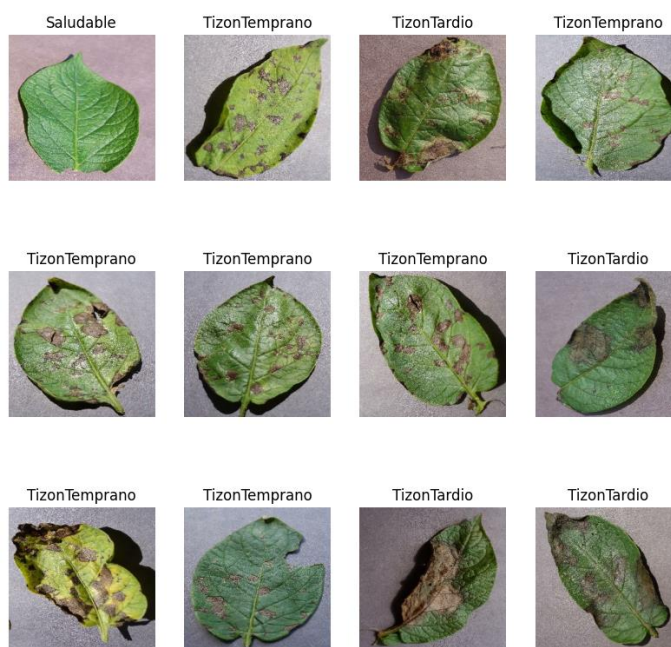


*Nota.* En la imagen se puede observar valores de RGB en forma matricial tomado de [deitm.edu.co](http://deitm.edu.co) | *Detección de enfermedades en plantas de crisantemo, por medio de visión artificial aplicada a imágenes multiespectrales* (Sánchez Ríos, 2020)

Mostramos un grupo de imágenes tomadas aleatoriamente de la “Database” ver Figura 46, visualizando las etiquetas con lo que vamos a trabajar:

### Figura 46

*Matriz de 3x4 de la base de datos*



*Nota.* Se muestra 12 imágenes tomadas de la “BaseData” con la etiqueta correspondiente a cada enfermedad

### Optimización de la memoria

Los tamaños de lote que cuyas potencias son de 2 permiten un uso más eficiente de la memoria y de los recursos de cálculo. Los procesadores modernos, especialmente aquellos diseñados para cálculos paralelos (como las GPU's utilizadas en el entrenamiento de redes neuronales), a funcionan de manera eficiente con este tipo de tamaños.

Podemos encontrar tamaños como 8, 16, 32, 64, 128, etc. Para este caso, extraemos la información de un lote de tamaño 32.

### **División de los datos de para el entrenamiento validación y prueba**

En este tipo de proyectos de inteligencia artificial, al trabajar con redes neuronales convolucionales para el procesamiento de imágenes, es común dividir el conjunto de datos en tres subconjuntos: entrenamiento, validación y prueba. La división está realizada como muestra la tabla 2.

**Tabla 2**

Porcentajes de división de datos para el modelo

| <b>Datos</b>  | <b>Porcentaje</b> | <b>Descripción</b>   |
|---------------|-------------------|--|
| Entrenamiento | 80%               | Se utiliza para entrenar el modelo de aprendizaje automático   |
| Validación    | 10%               | <i>Se utiliza para evaluar el rendimiento del modelo durante el entrenamiento y ajustar los hiperparámetros (como la tasa de aprendizaje).</i> |
| Prueba        | 10%               | Se utiliza para evaluar el rendimiento final del modelo después de que se ha completado el entrenamiento                                       |

Durante el entrenamiento, el modelo ajusta sus parámetros (pesos y sesgos en el caso de las redes neuronales) para minimizar el error entre sus predicciones y los datos reales. La validación ayuda a prevenir el sobreajuste, que ocurre cuando el modelo se ajusta demasiado a los datos de entrenamiento y pierde la capacidad de generalizar a nuevos datos. Y dado que el modelo no ha visto estos datos durante el entrenamiento o la validación, el conjunto de prueba



proporciona una estimación imparcial de cómo se comportará el modelo con datos nuevos y desconocidos.

### **Normalización de los datos**

Cuando normalizamos las imágenes, generalmente escalamos los valores de los píxeles para que estén en el rango de 0 a 1 de los rangos 0 a 255 que presentan inicialmente. Este proceso ayuda a homogeneizar los datos de entrada y a hacer que el entrenamiento del modelo sea más eficiente.

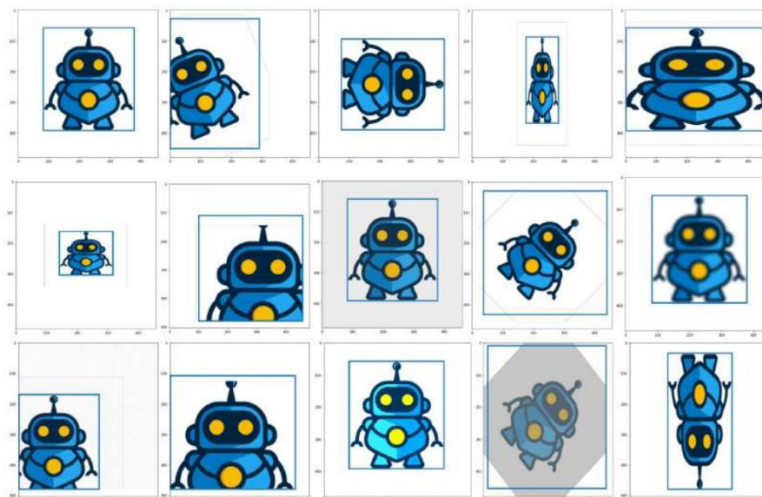
### **Aumento de datos (imágenes)**

Se implementó el 'ImageDataGenerator' que es una clase en la biblioteca Keras de TensorFlow utilizada para ejecutar preprocesamiento y aumentar los datos de imágenes. Esta clase permite "generar" de datos, para lo cual crea un objeto que genera lotes de imágenes que son transformadas a partir del conjunto original "BaseData" de imágenes en este caso de las 2152 imágenes de la planta de papa. Esto brinda una ventaja de rendimiento ya que las transformaciones se realizan en tiempo real en el hardware del computador, logrando ahorro de memoria, esto, al no ser necesario almacenar las versiones transformadas de las imágenes

El código usado fue 'rotation\_range', se utiliza para rotar aleatoriamente las imágenes dentro de un rango de grados dado, 'horizontal\_flip' se utiliza para girar aleatoriamente las imágenes de forma horizontal. En la figura 47 se observa un ejemplo de como, desde la imagen original se pueden crear varias imágenes, con rotaciones, giros, acercamientos, alejamientos, entre otros.

## Figura 47

Ejemplo visual de los tipos de aumento de datos



*Nota.* Se puede observar que la primera imagen es la original y las demás imágenes son las generadas automáticamente a través de transformaciones

de <http://sedici.unlp.edu.ar/> *Detección de enfermedades y plagas en cultivos mediante Machine Learning* (Pereyra, 2020)

## Entrenamiento

Las CNN's son adecuadas para tareas de visión por computadora y procesamiento de imágenes, en este caso al reconocer la enfermedad tizón tardío mediante imágenes y por consecuencia, al trabajar únicamente con imágenes las cuales contienen, patrones, bordes, texturas, colores y formas ocasionadas por la enfermedad. Se ha elegido el modelo. Además, el número de datos y la generación de datos nos permiten trabajar con dicho modelo.

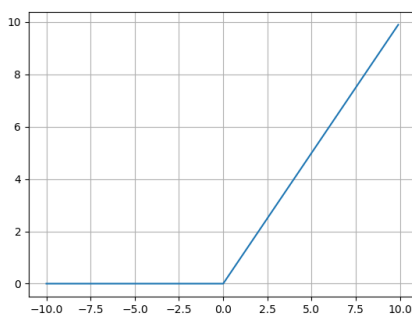
## Capas de convolución

Se utilizan varias capas convolucionales cada una con filtros diferentes para extraer características relevantes de las imágenes. Cada capa convolucional está seguida de una

función de activación ReLU, ver Figura 48, para introducir no linealidades en el modelo y permitir la detección de patrones más complejos.

### Figura 48

*Gráfica de la Activación ReLU*



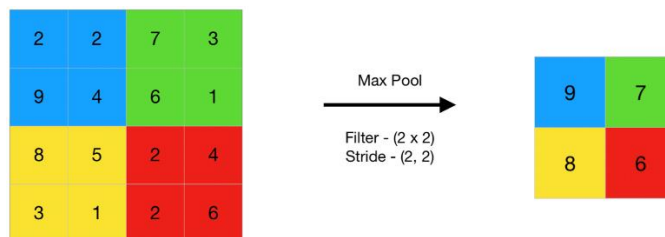
*Nota.* Se muestra la gráfica de la función ReLU tomada de Jahaziel Ponce (Ponce, 2021).

### Capas de reducción de dimensionalidad

Después de cada capa convolucional, se aplican capas de reducción de dimensionalidad en forma de Max Pooling (MaxPooling2D). Esto reduce el tamaño de las representaciones de las características, lo que ayuda a reducir la cantidad de parámetros y el costo computacional del modelo, además de permitir una mayor generalización. En la figura 49 se puede observar el ejemplo de manera gráfica de la reducción de dimensionalidad al tomar los valores máximos en cada arreglo.

### Figura 49

*Representación gráfica de la función MaxPooling2D*



*Nota.* Se muestra la gráfica de la función MaxPooling2D tomada de geeksforgeeks. (Saviak, 2020)

### Capa Flatten

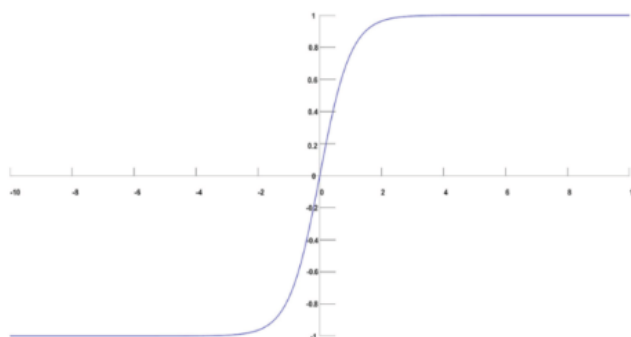
Al final de la secuencia de capas convolucionales y de reducción de dimensionalidad, se encuentra una capa Flatten que transforma los datos 2D en un vector unidimensional.

### Capas Densas

Después de la capa Flatten, se añaden dos capas densas (fully-connected) para realizar la clasificación final. La primera capa densa tiene 64 neuronas y utiliza la función de activación ReLU, mientras que la segunda capa densa tiene un número de neuronas igual al número de clases a predecir, en este caso tresy utiliza la función de activación Softmax ver Figura 50. Se ha seleccionado la función de activación Softmax para la última capa debido a que el problema es una clasificación multiclase con tres clases posibles: tizón tardío, tizón temprano y estado saludable. La salida del modelo será entonces un vector de tres probabilidades que suman 1, y la clase predicha será aquella con la mayor probabilidad.

### Figura 50

*Función SoftMax: Activación para la clasificación*



*Nota.* Gráfica de la función Softmax tomado de Jacar (Javi, 2023)

## Capítulo IV

### Pruebas

En este capítulo, se busca evaluar el desempeño, cambiando diferentes parámetros en la solución, con el fin de obtener la mejor confianza en las predicciones, y se observan los resultados de la solución final, se evalúan las imágenes de prueba separadas para el testeo, además se incluyen imágenes fuera del “DataBase” para verificar su predicción.

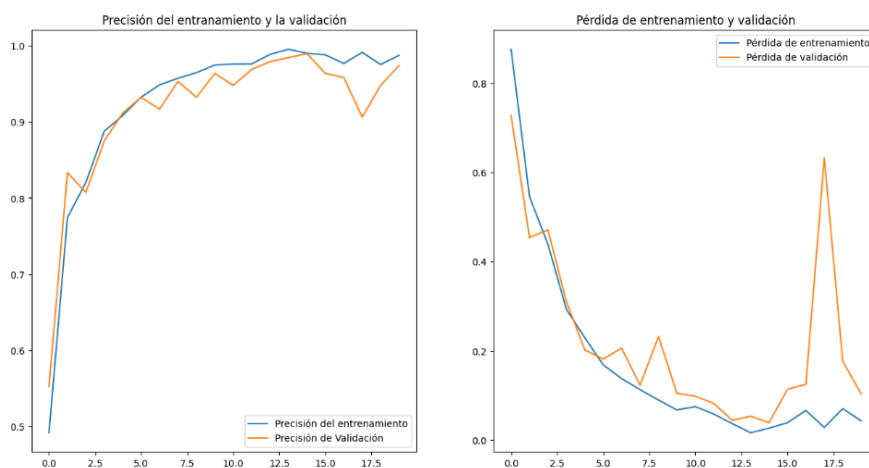
Comparación con aumento de datos y sin aumento de datos.

### Modelo sin aumento de datos

En la Figura52 se muestra la gráfica de la precisión y pérdida del entrenamiento y la validación, se observa que en la pérdida existe un pico, esto generalmente es un signo de sobreajuste. Esto significa que el modelo está comenzando a memorizar los datos de entrenamiento en lugar de generalizar correctamente.

### Figura 51

*Precisión y pérdida del entrenamiento y validación sin aumento de datos*



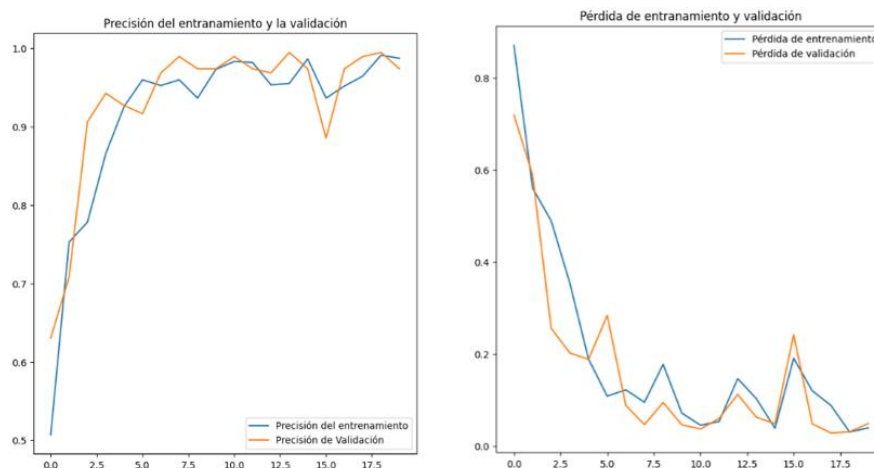
*Nota.* El gráfico representa como se comportan el entrenamiento versus la validación en la precisión y pérdida para los datos sin aumento

### Modelo con aumento de datos

En la Figura 52 se muestra la gráfica de la precisión y pérdida del entrenamiento y la validación, se observa que en la pérdida se estabiliza y ya no provoca el pico comparado con la anterior gráfica. Esto indica que el modelo ha aprendido adecuadamente el patrón en los datos y no muestra signos evidentes de sobreajuste, esta es la situación ideal. La pérdida de entrenamiento es ligeramente más baja que la pérdida de validación y ambas se han estabilizado, por lo tanto el modelo ha generalizado de manera correcta al conjunto de datos de validación.

### Figura 52

*Precisión y pérdida del entrenamiento y validación con aumento de datos*



*Nota.* El gráfico representa como se comportan el entrenamiento versus la validación en la precisión y pérdida pero con aumento de datos

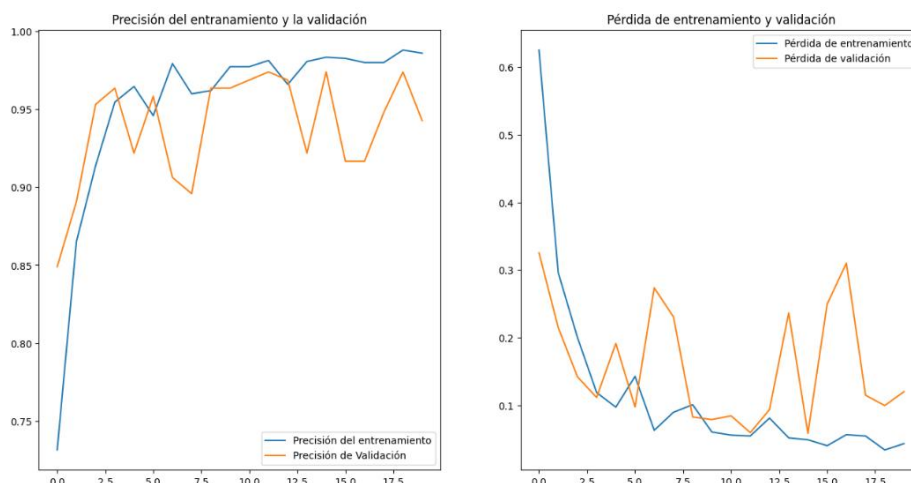
### Comparación del entrenamiento con el número de neuronas y épocas

A continuación, en el gráfico de la Figura 53 se muestra el entrenamiento con 32 neuronas situadas en la capa densa y 20 épocas, existe signo de sobreajuste. Es decir, el modelo puede

estar memorizando el conjunto de entrenamiento y perdiendo capacidad de generalización en datos no vistos.

### Figura 53

*Desempeño del modelo con 32 neuronas en la capa densa y 20 épocas*

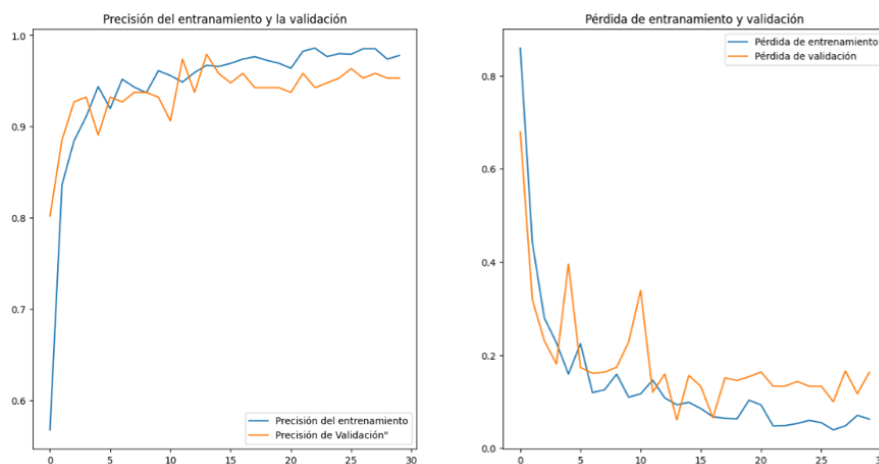


*Nota.* El gráfico representa como se comportan el entrenamiento versus la validación en la precisión y pérdida con 32 neuronas en la capa densa y 20 épocas.

La Figura 54 muestra 64 neuronas en la capa densa, una nueva capa convolucional y 30 épocas dando una mejora considerable en las fluctuaciones y estabilidad tanto en la precisión de la validación como en la pérdida de la validación, lo que indica que el modelo está correctamente ajustado, es decir, no sufre de un sobreajuste marcado. Entonces el modelo ha alcanzado un punto donde ha aprendido lo máximo posible dadas las características y datos actuales. Añadir más épocas de entrenamiento más allá de este punto probablemente no resultará en mejoras significativas y podría incluso llevar al modelo a un sobreajuste.

## Figura 54

*Desempeño del modelo con 64 neuronas en la capa densa y 30 épocas*



*Nota.* El gráfico representa como se comportan el entrenamiento versus la validación en la precisión y pérdida con 64 neuronas en la capa densa y 30 épocas.

## Resultados

Se busca evaluar el desempeño, confianza de la solución planteada, se evalúan las imágenes de prueba separadas para el testeo, además con imágenes fuera del "DataBase", es decir con imágenes diferentes a las recolectadas para el modelo.

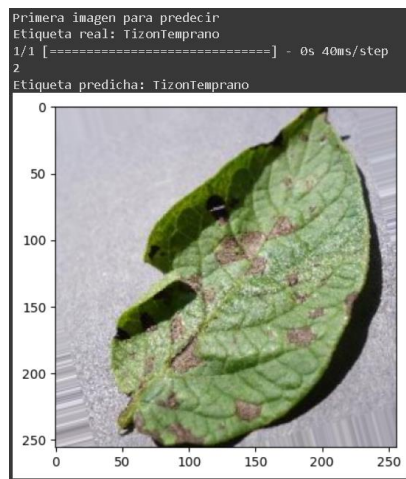
### Primera predicción

En la figura 55 se observa una primera ejecución del algoritmo de solución, observamos que es una imagen creada con el generador, donde la etiqueta real muestra que es el Tizón temprano, y la etiqueta predicha confirma que es una hoja infectada con el Tizón temprano, se puede ver que el tiempo requerido para predecir la imagen es de 40 m/s.



## Figura 55

*Primer ejemplo aplicando el modelo de solución*



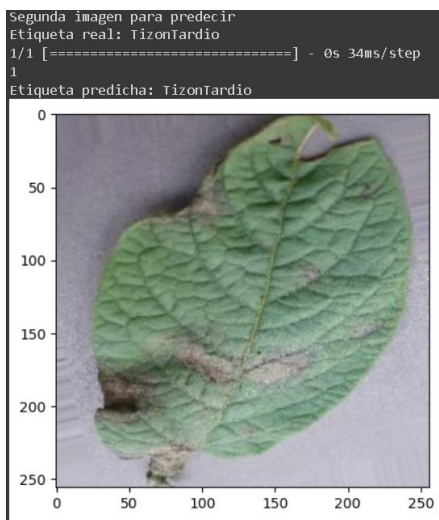
*Nota.* La imagen tomada muestra la etiqueta real y predicha de un ejemplo aleatorio del banco de pruebas junto con el tiempo que se demora, en este caso 40m/s

### Segunda predicción

En la figura 56 se observa una segunda ejecución del algoritmo de solución, observamos que al igual que el ejemplo de la figura 57, también es un imagen creada con el generador, donde la etiqueta real muestra que es el Tizón tardío, y la etiqueta predicha confirma que es una hoja infectada con el Tizón tardío, se puede ver que el tiempo requerido para predecir la imagen es de 34 m/s.

## Figura 56

Segunda ejemplo aplicando el modelo de solución



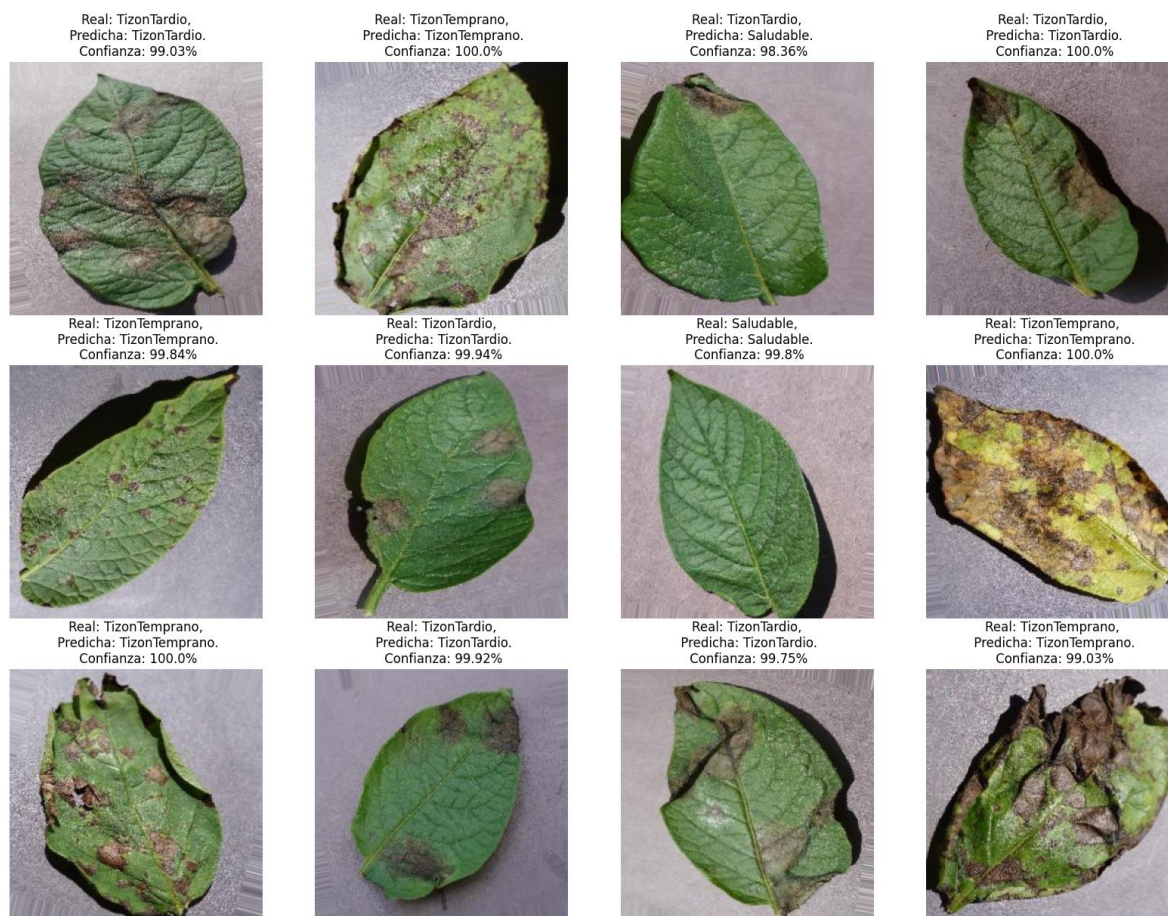
*Nota.* La imagen tomada muestra la etiqueta real y predicha de otro ejemplo aleatorio del banco de pruebas junto con el tiempo que se demora, en este caso 34m/s

### Confianza del modelo

La "confianza" es una medida de cuán seguro está el modelo de su predicción. En este caso observamos la probabilidad que el modelo asigne a la clase predicha. Cuan más alto es el valor de la confianza, más seguro está el modelo de su predicción, observamos que la confianza para este conjunto de 12, el porcentaje más bajo es del 98,36% y llegando hasta el 100%, lo cual corrobora el correcto funcionamiento del algoritmo de solución.

## Figura 57

Conjunto de imágenes aplicando el algoritmo de solución



*Nota.* La imagen tomada muestra una matriz de imágenes con la etiqueta real y predicha de un ejemplo aleatorio del banco de pruebas junto con el nivel de confianza

## Capítulo V

### Conclusiones

- Se diseñó un modelo de reconocimiento de la enfermedad tizón tardío y tizón temprano que infectan las hojas de la papa, se aplicó técnicas de aprendizaje profundo para la identificación de las enfermedades.
- El modelo de solución demuestra ser efectivo en la tarea de detección de la enfermedad estudiada tizón tardío, al procesar y clasificar las imágenes ingresadas al modelo de las tres clases, que son estados saludables hasta diferentes afectaciones de tizones. Los resultados indican la precisión y eficiencia del modelo generado, lo que evidencia que se debe combinar técnicas avanzadas de procesamiento de imágenes con Deep Learning para el mejor rendimiento.
- Al aplicar redes neuronales convolucionales (CNN) para la clasificación de enfermedades en las imágenes de la papa infecta, queda claro que son adecuadas para tareas de visión artificial, por los patrones, colores, texturas y formas que provocan los diferentes tizones.
- El cambio en los parámetros de generación de datos, épocas, número de neuronas, provoca cambios considerables en el rendimiento del modelo y los sobreajustes, por lo que ir variando los mismos, nos ayuda a encontrar el que mejor se adapte a la solución.

### Recomendaciones

- Se recomienda añadir más técnicas de aumento de datos como zoom, desplazamiento, giro vertical, izquierda, etc., esto ayudará a aumentar la variabilidad de las imágenes y reducir el sobreajuste, considerando que el uso del hardware aumentará.
- La mayoría de los datos, son imágenes de una sola hoja por cada dato, por lo que, si se desea aplicar el modelo en imágenes fuera del "DataBase" y que contengan la planta en sí, se debe separar las hojas y así tener el mismo contexto de dato, esto con el fin de una predicción correcta de la enfermedad.

## Referencias Bibliográficas

Abad, V. (2014). Evaluación del crecimiento y valor nutricional de la soya para forraje (glycine max) utilizando Biol como abono obtenido con microorganismos nativos. *Universidad de las Fuerzas Armadas ESPE. Carrera de Ingeniería en Biotecnología.*

Abood, M., Al-Rubaiee, S., & Khasawneh, M. (2019). *A survey on deep learning frameworks: TensorFlow, PyTorch, and Keras.* Journal of Big Data, pag:1.

Acuña, D. (2019). *Visión artificial aplicada a la detección e identificación de personas en tiempo real.* . Quito-Ecuador: EPN.

Acuña, I. (03 de 08 de 2023). *Portalfruticola.com.* Obtenido de <https://www.portalfruticola.com/noticias/2022/09/08/manejo-integrado-del-tizon-tardio-de-la-papa/>

Acuña, I., & Gutiérrez, M. (2004). *tizon.inia.cl.* Obtenido de <https://tizon.inia.cl/assets/boletines/47-como%20reconocer%20los%20tizones%20de%20la%20papa.pdf>

Acuña, I., Sandoval, C., & Sepúlveda, C. (2021). *Enfermedades de la papa: Plataforma de evaluación de riesgo sanitario.* Chile: Instituto de Investigaciones Agropecuarias INIA.

Aggarwal, C. (2014). *Data mining: the textbook.* Springer.

Agrobiotecnología. (12 de 10 de 2017). *agroavances.com.* Obtenido de <https://agroavances.com/noticias-detalle.php?idNot=1020>

Alam, T., & Rustgi, S. (03 de 08 de 2023). *eOrganic.org.* Obtenido de <https://eorganic.org/node/35266>

Amat Rodrigo, J. (04 de 2017). *cienciadedatos.net*. Obtenido de [https://cienciadedatos.net/documentos/34\\_maquinas\\_de\\_vector\\_soporte\\_support\\_vector\\_machines](https://cienciadedatos.net/documentos/34_maquinas_de_vector_soporte_support_vector_machines)

Amorim, R., Nguyen, H., Mogesen, P., Kovács, I., Wigard, J., & Sorensen, T. (2017). *IEEE Wireless Communications Letters*. IEEE 6th ed., pags. 514-517.

Angulo, C., & etal. (2012). Compensación de los efectos generados en la imagen por el control de navegación del robot Aibo ERS 7. *VII Congreso de Ciencia y Tecnología ESPE*.

Angulo, C., & etal. (2014). Real-time video stabilization without phantom movements for micro aerial vehicles. *EURASIP Journal on Image and Video Processing*, 1, 1-13.

Aprende Machine learning. (06 de 06 de 2021). Obtenido de [Aprendemachinelarning.com: https://www.aprendemachinelarning.com/como-funcionan-lasconvolutional-neural-networks-vision-por-ordenador/](https://www.aprendemachinelarning.com/como-funcionan-lasconvolutional-neural-networks-vision-por-ordenador/)

Arifin, F., Arifandi, D., & Didit, W. (2014). *Autonomous detection and tracking of an object autonomously using AR.Drone quadcopter*. *Journal of Computer Science and Information*, 7(7), págs.11-17.

Asimbaya, I., Albán, P., Fernández, Y., & etal. (2020). Ackermann UGV with 2D Mapping for Unknown Environments. En *International Conference of Research Applied to Defense and Security* (págs. 45-53).

Barrios, J. (22 de 04 de 2022). *www.juanbarrios.com*. Obtenido de <https://www.juanbarrios.com/redes-neurales-convolucionales/>

Basantes, J., Godoy, L., Carvajal, T., Castro, R., Toulkeridis, T., & etal. (2018). Capture and processing of geospatial data with laser scanner system for 3D modeling and virtual reality of Amazonian Caves. *IEEE Ecuador Technical Chapters Meeting (ETCM)*. Samborondón, Ecuador.

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2006). *Speeded-up robust features (SURF)*. Computer vision and image understanding, pags: 346-359.

Bell, A., & Sejnowski, T. (1997). *The independent components of natural images*. Neural computation, pags:1129-1159.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Breiman, L. (2001). *Random forests*. Machine learning, pags:5-32.

Caballeros, J., Segarra, D., Castro, P., & etal. (2018). RRT Path Planning and Morphological Segmentation Based Navigation for a Tetrapod Robot. *Ubiquitous Networking: 4th International Symposium, UNet* (págs. 273-280). Springer International Publishing.

Cabras, P., Rosell, J., Pérez, A., Rosell, A., & etal. (2011). Haptic-based navigation for the virtual bronchoscopy. *18th IFAC World Congress*. Milano, Italy.

Caiza, O., Albán, P., Fernández, Y., & etal. (2020). Kinect and Manipulator-Based Sample Collection System for Military Robot. En *Smart Innovation, Systems and Technologies* (págs. 75-87).

Calderón, M., Merizalde, D., & etal. (2020). Visual-Based Real-Time Detection Using Neural Networks and Micro-UAVs for Military Operations. En *Smart Innovation, Systems and Technologies* (págs. 55-64).

Campos Soberanis, M. (13 de 02 de 2018). *Medium.com*. Obtenido de <https://medium.com/soldai/inspiraci%C3%B3n-biol%C3%B3gica-de-las-redes-neuronales-artificiales-9af7d7b906a>

Carmona, E. (2014). *Tutorial sobre Máquinas de Vectores Soporte (SVM)*. UNED.

Carrera, A., Merizalde, D., Calderón, M., & etal. (2020). Monte Carlo-Based Localization for Kidnapped Robot Problem. En *International Conference of Research Applied to Defense and Security* (págs. 101-109).

Casaliglla, V., Pólit, J., & etal. (2017). Obstacle Avoidance Based-Visual Navigation for Micro Aerial Vehicles. *Electronics* , 6 (1), 10.

CEPAL, & Rodrigues, M. (2015). *Penetración de las TIC en la agricultura y las zonas rurales de América Latina: Estimaciones e impactos*. Seminario Internacional TIC (1.a ed., Vol. 1). III.

Chávez, M. (2018). *Mejoramiento de la productividad mediante la implementación de drones en el cultivo de flores de verano y rosas en el Grupo Esmeralda Ecuador*. Quito: Pontificia Universidad Católica del Ecuador, (1.a ed., Vol. 1).

Chmaj, G., & Selvaraj, H. (2015). *istributed processing applications for UAV/drones: a survey*. Springer.

Chollet, F. (2015). *Keras*. GitHub repository.

Chollet, F. (2015). *Keras: The Python Deep Learning library*. Obtenido de <https://keras.io/>

Cobeña, B., Rodriguez, G., Salcedo, V. S., Collaguazo, B., & etal. (2018). SVM and RGB-D Sensor Based Gesture Recognition for UAV Control. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics* (págs. 713-719). Springer.

Collaguazo, B., Tituaña, J., Amaguaña, F., & etal. (2018). Simulation System Based on Augmented Reality for Optimization of Training Tactics on Military Operations. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics* (págs. 394-403). Springer.

*Communication and coordination for drone networks*2017New YorkNY: Springer.



Conci, A., Falconi, L. G., Perez, M., & etal. (2020). Transfer learning and fine tuning in breast mammogram abnormalities classification on CBIS-DDSM database. *Adv. Sci. Technol. Eng. Syst. J* , 154-165.

Córdova, A., Jiménez, B., Pardo, J., Toalombo, I., & etal. (2018). Geolocation and Counting of People with Aerial Thermal Imaging for Rescue Purposes. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics* (págs. 171-182). Springer.

Cortes, C., & Vapnik, V. (1995). *Support vector networks*. Machine Learning, pags:273-297.

Costa, J., Goulart, V., & Fernandes, C. (2020). *Detección temprana de tizón tardío en cultivos de papa mediante aprendizaje supervisado*. Revista de Agricultura de Precisión.

Costa-Castello, R., Angulo, C., & etal. (2017). Autonomous Navigation Control for Quadrotors in Trajectories Tracking. En *Lecture Notes in Computer Science* (págs. 287-297).

Cover, T., & Hart, P. (1967). *Nearest neighbor pattern classification*. IEEE transactions on information theory, pags: 21-27.

Cruz, H., Meneses, J., Aguilar, W., & Andrade-Miranda, G. (2019). Detection and Segmentation of Ecuadorian Deforested Tropical Areas Based on Color Mean and Deviation. *Information Technology and Systems: Proceedings of ICITS* (págs. 452-461). Springer International Publishing.

Dalal, N., & Triggs, B. (2005). *Histograms of oriented gradients for human detection*. IEEE Transactions on Pattern Analysis and Machine Intelligence, pags:69-88.

De la Rosa, F. (2016). *Aplicación de Deep Learning en Robótica Móvil para Exploración y Reconocimiento de Objetos basados en Imágenes*. Colombia: IEEE.

Dirección General de Sanidad, Centro de Nacional de Referencia Fitosanitaria. (2014). *Protocolo de Diagnóstico Potato Virus Y (PVYN y PVYNTN)*. Mexico: Sagarpa.

Duque Domingo, J. (18 de 03 de 2022). *Cartif*. Obtenido de <https://blog.cartif.es/deep-learning-vision-artificial/>

El comercio-pe. (03 de 08 de 2023). *Suplementos Peru*. Obtenido de <http://suplementos.ec.pe/suplementos/comercial/gastronomia/que-debemos-consumir-papa-1003026/m>

El Universo. (30 de JUNIO de 2019). Una enfermedad que mata a la palma aceitera deja desempleo en Ecuador. Quito, Pichincha, Ecuador.

Falconí, L. G., Pérez, M., & etal. (2019). Transfer learning in breast mammogram abnormalities classification with mobilenet and nasnet. *International Conference on Systems, Signals and Image Processing (IWSSIP)*, (págs. 109-114).

Falconi, L. G., Perez, M., Conci, A., & etal. (2020). Transfer learning and fine tuning in mammogram bi-rads classification. *IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS)*, (págs. 475-480).

Felzenszwalb, P., Huttenlocher, D., & Belongie, S. (2004). *Pictorial structures for object recognition*. *International journal of computer vision*, pags:55-79.

Fernández Castilla, N. (2013). *Visión Artificial Avanzada. Introducción a la Visión Artificial*. pags:1-50.

Fleet, D., & Viola , T. (2001). *Image Features and Learning for Visual Recognition*. IEEE.

Food Security. (2018). *Potato production in sustainable soil management systems*. Vol.10.

Forsyth, D., & Ponce , J. (2019). *Computer Vision: A Modern Approach*.

Galarza, J., Pérez, E., Serrano, E., Tapia, A., & et al. (2018). Pose estimation based on monocular visual odometry and lane detection for intelligent vehicles. En *Lecture Notes in Computer Science* (págs. 562-566).

Galindo, R., Reyes, R., & et al. (2019). Landmark based eye ratio estimation for driver fatigue detection. En *Lecture Notes in Computer Science* (págs. 565-576). Germany: Springer.

Gonzalez , R., & Woods , R. (2018). *Digital Image Processing*.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Grijalva, S., & et al. (2019). Landmark-Based Virtual Path Estimation for Assisted UAV FPV Tele-Operation with Augmented Reality. En *Lecture Notes in Computer Science* (págs. 688-700).

Grijalva, S., Cueva, G., Ramírez, D., & et al. (2019). HSVM-Based Human Activity Recognition Using Smartphones. En *Lecture Notes in Computer Science* (págs. 217-228).

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning*. New York: Springer-Verlag.

Haykin, S. (1994). *Neural networks: A comprehensive foundation*. New York: Macmillan.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep residual learning for image recognition*. arXiv preprint arXiv:1512.03385.

Hinton , G., & Sejnowski , T. (1986). *A Survey of Artificial Neural Networks*.

Hortalizas. (03 de 08 de 2023). *hortalizas.com*. Obtenido de <https://www.portalfruticola.com/noticias/2019/08/12/pudricion-blanda-de-la-papa-el-patogeno-que-mas-afecta-a-este-cultivo/>

Huang , F., Bottou , L., & LeCun, Y. (2004). *Convolutional Neural Networks for Visual Recognition*. IEEE.

*Huertos y mas.* (03 de 08 de 2023). Obtenido de <https://huertosymas.com/etapas-de-crecimiento-de-la-papa/>

2017 *ImageNet Large Scale Visual Recognition Challenge (ILSVRC)* Princeton-Usa Universidad de Stamford.

image-net.org. (2015). *image-net*. Obtenido de <https://www.image-net.org/about.php>

INIAP Estación Experimental Santa Catalina. (JULIO de 2015). *VI Congreso ecuatoriano de la papa*. Obtenido de <https://repositorio.iniap.gob.ec/bitstream/41000/3061/1/iniapsc332p.pdf>

International Potato Center . (2023). *The potato as a global food security crop*. Lima: CIP.

International Potato Center. (04 de 08 de 2023). *cipotato.org*. Obtenido de [https://cipotato.org/es/potato/potato-facts-and-figures/#:~:text=La%20papa%20\(Solanum%20tuberosum\)%20pertenece,del%20arroz%20y%20del%20trigo.](https://cipotato.org/es/potato/potato-facts-and-figures/#:~:text=La%20papa%20(Solanum%20tuberosum)%20pertenece,del%20arroz%20y%20del%20trigo.)

Jain , A., & Vailaya , A. (1999). *A Survey of Image Feature Extraction Techniques*. IEEE.

Jara-Olmedo, A., Medina-Pazmiño, W., Mesías, R., Araujo-Villaroel, B., Pardo, J., & etal. (2018). Interface of Optimal Electro-Optical/Infrared for Unmanned Aerial Vehicles. En *Smart Innovation, Systems and Technologies* (págs. 372-380).

Jara-Olmedo, A., Medina-Pazmiño, W., Tozer, T., Pardo, J., & etal. (2018). E-services from Emergency Communication Network: Aerial Platform Evaluation. *International Conference on eDemocracy & eGovernment (ICEDEG)* (págs. 251-256). IEEE.

Javi. (Marzo de 2023). *Jacar*. Obtenido de <https://jacar.es/funcion-softmax-activacion-para-la-clasificacion/#:~:text=La%20funci%C3%B3n%20SoftMax%20es%20una%20funci%C3%B3n%20de%20activaci%C3%B3n%20que%20se, en%20probabilidades%20que%20suman%20uno.>

Jayme, G. (2016). A review on the main challenges in automatic plant disease identification based on visible range images. *Sciencedirect* , 52-60.

Journal of Soil Science and Plant Nutrition. (2015). A review of soil management practices for potato production. Vol 55.No2.

kaggle. (s.f.). *kaggle*. Obtenido de <https://www.kaggle.com/>

keepcoding. (17 de 08 de 2022). *keepcoding.io*. Obtenido de <https://keepcoding.io/blog/funcion-de-activacion-en-deep-learning/>

Krizhevsky, A., Sutskever , I., & Hinton , G. (2012). *Deep Convolutional Neural Networks for Large-Scale Visual Recognition*.

Kurtulmus, F., & Kavdir, I. (2014). *Detecting corn tassels using computer vision and support vector machines*. Elsevier.

Labra, C. (2021). *¿Cómo iniciarse en la inteligencia artificial?* Head Centro de Excelencia Datos & IA.

LearnOpenCV. (6 de 12 de 2016). *LearnOpenCV.com*. Obtenido de <https://learnopencv.com/histogram-of-oriented-gradients/>

LeCun, Y., Bengio, Y., & Hinton, G. (2015). *Deep learning*. *Nature*. pags, 436-444.

León, M., Arroyo, J., & Fuentes, A. (2019). *Detecção de tizón tardío em cultivo de batata usando aprendizagem profunda*. Brasil: In Anais do VII Brazilian Workshop on Computer Vision, pags:61-68.

Lienhart, R., Maydt, J., Huiskes, D., & Cremers, D. (2002). *An extended set of Haar-like features for rapid object detection*. *International Journal of Computer Vision*, pags:177-195.

Limaico, A., Rodríguez, G., Quisaguano, F., Sandoval, D., Caballeros, J., & etal. (2018). Graph Based RRT Optimization for Autonomous Mobile Robots. *Intelligence Science and Big*

*Data Engineering: 8th International Conference, IScIDE* (págs. 12-21). Springer International Publishing.

Lowe, D. J. (2004). *Vision*.

Luna, M. A., Moya, J. F., Abad, V., Ruiz, H., Parra, H., & etal. (2017). Pedestrian Detection for UAVs Using Cascade Classifiers and Saliency Maps. En *Lecture Notes in Computer Science* (págs. 563-574).

Luna, M. A., Ruiz, H., Moya, J. F., Luna, M. P., Abad, V., & etal. (2017). Statistical Abnormal Crowd Behavior Detection and Simulation for Real-Time Applications. En *Lecture Notes in Computer Science* (págs. 671-682).

MacQueen, J. (1967). *Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1, pags: 281-297.

Manosalvas, J. F., Guillén, J. A., Collaguazo, B., & etal. (2018). Robust Motion Estimation Based on Multiple Monocular Camera for Indoor Autonomous Navigation of Micro Aerial Vehicle. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics* (págs. 547-561). Springer.

*Mastering the game of Go with deep neural networks and tree search*2016Nature, pags:484-489

McCulloch, S., & Pitts, W. (1943). *A logical calculus of the ideas immanent in nervous activity*. . *Bulletin of Mathematical Biophysics*, pags: 115-133.

Méndez, L. (2012). *Principales enfermedades que afectan al cultivo de papa*. INIA: INIA Carillanca, pags.7-20.

Merizalde, D., Calderón, M., & et al. (2020). Autonomous Navigation Based on Proportional Controller with GPS Setpoint for UAV in External Environments. En *Smart Innovation, Systems and Technologies* (págs. 89-99).

Mikolajczyk, K., & Schmid, C. (2005). *A performance evaluation of local descriptors*. IEEE transactions on pattern analysis and machine intelligence, págs:1615-1630.

Moposita, J., Orbea, D., Paredes, M., Reyes, R., Montoya, L., & et al. (2017). Vertical take off and landing with fixed rotor. *Chilean Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*. Pucón, Chile.

Morales, S., & et al. (2016). 3D Environment Mapping Using the Kinect V2 and Path Planning Based on RRT Algorithms. *Electronics* , 5 (4), 70.

Mordvintsev, A., & Abid , K. (2013). *Opencv:Face Detection using Haar Cascades*. Opencv.

Moya, J. F., Abad, V., Parra, H., Ruiz, H., Luna, M. A., & et al. (2017). Pedestrian detection for UAVs using cascade classifiers with meanshift. *IEEE 11th international conference on semantic computing (ICSC)*.

Murphy, K. (2012). *Machine learning: a probabilistic perspective*. MIT press.

Nasimba, J. (2017). *Estimación de la altura de columnas explosivas en el volcán Tungurahua basado en procesamiento de imágenes visuales y térmicas*. Riobamba: ESPOCH.

Nielsen, M. (2015). *How the backpropagation algorithm works*. En M. Nielsen, *Neural Networks and Deep Learning*. Usa: págs. 39-57.

Nvidia. (04 de 08 de 2023). *Nvidia.com*. Obtenido de <https://www.nvidia.com/es-la/glossary/data-science/computer-vision/>

Oliveros Sabogal, D. S. (2019). *Algoritmo para la medición del grado de severidad de tizón temprano causado en hojas de tomate, en análisis digital de imágenes*. Bogota D.C: Universidad del bosque.

Orbea, D., Moposita, J., Paredes, M., León, G., Jara-Olmedo, A., & etal. (2017). Math Model of UAV Multi Rotor Prototype with Fixed Wing Aerodynamic Structure for a Flight Simulator. En *Lecture Notes in Computer Science* (págs. 199-211).

Pardo, C. (04 de 08 de 2023). *carlosjuliopardoblog.wordpress.com*. Obtenido de <https://carlosjuliopardoblog.wordpress.com/2017/05/12/filtros-haar-deteccion-de-rostros/>

Pardo, J. (2016). *Sistema de comunicación inalámbrico para la transmisión de imágenes térmicas desde una aeronave pilotada remotamente a su estación de control en tierra*. UTCiencia, págs. 211-223.

Pardo, J., Toulkeridis, T., & etal. (2017). Wireless communication system for the transmission of thermal images from a UAV. *Chilean Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*. Pucón, Chile.

Pereyra, M. E. (Agosto de 2020). <http://sedici.unlp.edu.ar/>. Obtenido de [http://sedici.unlp.edu.ar/bitstream/handle/10915/118496/Documento\\_completo.pdf?sequence=1&isAllowed=y](http://sedici.unlp.edu.ar/bitstream/handle/10915/118496/Documento_completo.pdf?sequence=1&isAllowed=y)

Pérez, W., & Forbes, G. (2008). *Manual Técnico, El tizón tardío de la papa*. Lima, Peru: Centro Internacional de la Papa (CIP).

Pitarque, A., Roy, J., & Ruiz, J. (DICIEMBRE de 2018). *Universidad de Valencia*. Obtenido de Redes neurales vs modelos estadísticos: Simulaciones sobre tareas de predicción y clasificación (No. 19).: <https://www.uv.es/PSICOLOGICA/articulos3.98/pitarque.pdf>

plantvillage.psu.edu. (2013). *PlantVillage*. Obtenido de <https://plantvillage.psu.edu/>



Ponce, J. (Julio de 2021). *jahazielponce.com*. Obtenido de <https://jahazielponce.com/funciones-de-activacion-y-como-puedes-crear-la-tuya-usando-python-r-y-tensorflow/>

*Principios de manejo de control químico del tizón tardío de la papa en Ecuador* 2015 Ibarra Universidad Técnica del Norte, pag:91-93

Python Sitio Web Oficial. (junio de 13 de 2021). *python.org*. Obtenido de [www.python.org](http://www.python.org)

Quigley, M., Gerkey, B., Conley, K., Faust, J., Foote, T., & et al. (2010). *ROS: an open-source Robot Operating System*. USA: Computer Science Department, Stanford University, Stanford, CA.

Quinlan, J. (1986). *Induction of decision trees*. Machine learning, pags: 81-106.

Quisaguano, F. J., Alvarez, L. G., Pardo, J. A., Zahira, P., & et al. (2018). Monocular depth perception on a micro-UAV using convolutional neuronal networks. En *Lecture Notes in Computer Science* (págs. 392-397).

Quisaguano, F. J., Rodríguez, G. A., Alvarez, L. G., Limaico, A., S, S. D., & et al. (2018). Convolutional neuronal networks based monocular object detection and depth perception for micro UAVs. En *Lecture Notes in Computer Science* (págs. 401-410).

Ramírez, Ó., & et al. (2019). Conciencia situacional en la frontera, proporcionada por sensores de alta tecnología. *Revista de Ciencias de Seguridad y Defensa* , 169-178.

Ramos Alonso, J. A. (2016). *Generacion de algoritmos para clasificacion de obstaculos*. Madrid: Universidad Carlos III.

Reyes, R., Pérez, H., Paredes, M., Montoya, L., & et al. (2017). MilNova: An approach to the IoT solution based on model-driven engineering for the military health monitoring. *CHILEAN*

*Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)* (págs. 1-5). IEEE.

Rodríguez, G. A., Álvarez, L., Sandoval, S., Quisaguano, F., Limaico, A., & etal. (2017). On-Board Visual SLAM on a UGV Using a RGB-D Camera. En *Lecture Notes in Computer Science* (págs. 298-308).

Rodríguez, V. (2018). *Conceptos básicos sobre redes neuronales*. Personal.

Rojas, I., Álvarez, L., Grijalva, S., & etal. (2019). Monocular Vision-Based Dynamic Moving Obstacles Detection and Avoidance. En *Lecture Notes in Computer Science* (págs. 386-398).

Rosenblatt, F. (1958). *The perceptron: A probabilistic model for information storage and retrieval*. *Psychological Review*, pags: 386-408.

Russell, S., & Norvig, P. (2020). *Artificial intelligence: A modern approach*. Pearson Education, (3.a ed.).

Salazar, J. (s.f.). *REDES INALÁMBRICAS*. Obtenido de [https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01\\_R\\_ES.pdf](https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01_R_ES.pdf)

Salcedo, V. S., Cobeña, B., Pardo, J. A., Zahira, P., & etal. (2018). On-board target virtualization using image features for UAV autonomous tracking. En *Lecture Notes in Computer Science* (págs. 384-391).

Salcedo, V., Sandoval, D., Cobeña, B., & etal. (2017). Developing of a Video-Based Model for UAV Autonomous Navigation. *Communications in Computer and Information Science* , 94-105.

Salgado, M. F., Tierra, A., Sandoval, D., & etal. (2017). Travel Time Estimation of Public Transport Networks Based on Commercial Incidence Areas in Quito Historic Center. *Analysis* .

Sánchez Ríos, C. A. (2020). Detección de enfermedades en plantas de. *Instituto Tecnológico Metropolitano* , 31.

Sanchez, E., & etal. (2019). Apoyo logístico de FF.AA ante un desastre natural. *Revista de Ciencias de Seguridad y Defensa* , 23-33.

Sandoval, S., Limaico, A., Villegas-Pico, M., Asimbaya, I., & etal. (2019). Path planning based navigation using LIDAR for an Ackerman unmanned ground vehicle. En *International Conference on Intelligent Robotics and Applications* (págs. 399-410).

Saviak, H. (2020). *Geeksforgeeks*. Obtenido de <https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/#article-meta-div>

Segarra, D., Caballeros, J., & etal. (2018). Visual based autonomous navigation for legged robots. *Intelligence Science and Big Data Engineering: 8th International Conference, IScIDE* (págs. 22-34). Springer International Publishing.

Segarra, D., Caballeros, J., Samà, A., Rodríguez-Martín, D., & etal. (2019). Orientation Estimation Using Filter-Based Inertial Data Fusion for Posture Recognition. En *Lecture Notes in Computer Science* (págs. 220-233).

Seongah, J., Simeone, O., & Joonhyuk, K. (2017). *Mobile cloud computing with a UAV-mounted cloudlet: optimal bit allocation for communication and computation (No. 27)*. Institution of Engineering and Technology. doi:10.1049/iet-com.2016.1114.

Sharifara, A; Mohd Rahim, M; Anisi, Y. (2014). *A general review of human face detection including a study of neural networks and Haar feature-based cascade classifier in face detection*. Kuala Lumpur, Malaysia, págs. 73-78.: 2014 International Symposium on Biometrics and Security Technologies (ISBAST).

siliconvalley. (16 de 04 de 2021). *siliconvalley.basisindependent.com*. Obtenido de <https://siliconvalley.basisindependent.com/2021/04/16/week-8-creating-initial-random-forest-model/>

Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv:1409.1556.

Singh, A., & Singh, D. (2021). *Potato Late Blight Disease Detection Using Convolutional Neural Networks*. International Journal of Advanced Research in Computer Science, pags:49-54.

Sutton, R., & Barto, A. (2018). *Reinforcement learning: An introduction*. MIT Press, (2.a ed.).

Szeliski, R. (2010). *Image Processing and Computer Vision*".

Tenezaca, D., Canchignia, C., Mendoza, D., & etal. (2019). Implementation of Dubin Curves-Based RRT\* Using an Aerial Image for the Determination of Obstacles and Path Planning to Avoid Them During Displacement of the Mobile Robot. En *Smart Innovation, Systems and Technologies* (págs. 205-215).

2016 *TensorFlow: Large-scale machine learning on heterogeneous systems* Google

2020 *The Difference Between Artificial Intelligence, Machine Learning and Us* Intel

Tierra, A., F, S. M., & etal. (2017). Travel Planning in Public Transport Networks Applying the Algorithm A\* for Metropolitan District of Quito. *Analysis* .

Tinizaray, P., Lucio, J., & etal. (2022). Fast segmentation of point clouds using a convolutional neural network for helping visually impaired people find the closest traversable region. *Inteligencia Artificial* , 50-63.

Tipanluisa, G. (08 de Noviembre de 2019). *El Comercio*. Obtenido de <https://www.elcomercio.com/cartas/agricultura-innovadora-ecuador.html>

Tituaña, J., Amaguaña, F., Calderón, M., Merizalde, D., & etal. (2020). Visual and Inertial Data-Based Virtual Localization for Urban Combat. En *International Conference of Research Applied to Defense and Security* (págs. 65-74).

Tiuma, A., Figueroa, J., Albán, P., Cruz, H., & etal. (2020). Low cost multitask mobile robot for military application. En *XV Multidisciplinary International Congress on Science and Technology* (págs. 462-473).

*Tuberculosis.org*

van Rossum, G. (1991). *Python Programming Language*. Obtenido de <https://docs.python.org/3/>

Vapnik, V. (1995). *The nature of statistical learning theory*. New York: Springer-Verlag.

Vargas, C. (03 de 08 de 2023). *Trycore*. Obtenido de <https://trycore.co/transformacion-digital/aplicaciones-machine-learning-en-organizaciones/>

Viola, P., & Jones, M. (2001). *Algoritmo de Viola-Jones*. Compaq.

Wang, L., & Han, J. (2014). *A survey on histogram of oriented gradients*. IEEE Transactions on Image Processing, pags: 3606-3622.

World Health Organization. (2022). *Potatoes: A healthy and sustainable crop for the 21st century*. WHO: Ginebra.

Zeiler, M., & Fergus, R. (2013). *Visualizing and Understanding Convolutional Networks*. New York-Usa: U. Cornell.

Zhang, J., & Zhang, Z. *A comprehensive survey on visual object detection*. IEEE Transactions on Pattern Analysis and Machine Intelligence, pags:2187-2204.

Zurita, M., Enríquez, V., & etal. (2019). Toward the Development of Surveillance and Reconnaissance Capacity in Ecuador: Geolocation System for Ground Targets Based on an Electro-Optical Sensor. En *Smart Innovation, Systems and Technologies* (págs. 123-133).