

**ANÁLISIS DE IMÁGENES MULTIESPECTRALES OBTENIDAS EN CULTIVOS  
DE ARROZ PARA LA DETERMINACIÓN DE ANOMALÍAS.**

**BRAYAN ANDRÉS ORTIZ GONZÁLEZ.  
EMANUEL HERNÁNDEZ MEDINA.**

**UNIVERSIDAD PONTIFICIA BOLIVARIANA  
ESCUELA DE INGENIERÍAS Y ARQUITECTURA  
FACULTAD DE INGENIERÍA ELECTRÓNICA  
MONTERÍA  
2022**

**ANÁLISIS DE IMÁGENES MULTIESPECTRALES OBTENIDAS EN CULTIVOS  
DE ARROZ PARA LA DETERMINACIÓN DE ANOMALÍAS.**

**BRAYAN ANDRÉS ORTIZ GONZÁLEZ  
EMANUEL HERNÁNDEZ MEDINA.**

**Trabajo de grado para optar al título de ingeniero electrónico**

**Director**

**CLAUDIA MILENA SERPA IMBETT, PhD  
Docente Interno Ing. Electrónica**

**UNIVERSIDAD PONTIFICIA BOLIVARIANA  
ESCUELA DE INGENIERÍAS Y ARQUITECTURA  
FACULTAD DE INGENIERÍA ELECTRÓNICA  
MONTERÍA  
2022**

## **LISTA DE CONTENIDO**

RESUMEN: .....	7
ABSTRACT: .....	8
1. INTRODUCCIÓN:	9
2. MARCO TEÓRICO/ESTADO DEL ARTE.	10
2.1 Marco de antecedentes. ....	10
2.2 Marco teórico. ....	11
2.2.1. Índices de vegetación. ....	11
2.2.2 Máquina de soporte vectorial. ....	13
2.2.3 Redes neuronales convolucionales. ....	16
3. MÉTODOS Y MATERIALES.	19
3.1 Sistema de alarma. ....	19
3.1.1 NDVI. ....	19
3.2 Algoritmos de aprendizaje (Deep Learning). ....	22
3.2.1 Red Neuronal Convolucional. ....	22
3.2.2 Máquina de soporte vectorial. ....	29
4. RESULTADOS Y DISCUSIÓN.	31
5. CONCLUSIONES Y RECOMENDACIONES.	37

6. BIBLIOGRAFÍA.	39
7. ANEXOS.	43

**LISTA DE FIGURAS**

Figura 1 .....	13
Figura 2 .....	15
Figura 3 .....	18
Figura 4 .....	19
Figura 5 .....	20
Figura 6 .....	21
Figura 7 .....	21
Figura 8 .....	22
Figura 9 .....	23
Figura 10 .....	24
Figura 11 .....	25
Figura 12 .....	26
Figura 13 .....	27
Figura 14 .....	27
Figura 15 .....	28
Figura 16 .....	29
Figura 17 .....	31
Figura 18 .....	32
Figura 19 .....	33
Figura 20 .....	34
Figura 21 .....	34

Figura 22 ..... 35

Figura 23 ..... 35

**RESUMEN:**

En este proyecto se desarrolló e implementó una modificación en el sistema de procesamiento de imágenes multiespectrales en Matlab AppDesigner que permite alertar al usuario el estado del cultivo de una zona segmentada mediante imágenes provenientes de cámaras multiespectrales o imágenes satelitales que calculan el valor de NDVI (Diferencia de Índice de Vegetación Normalizada) basada en medidas de la intensidad de la luz roja e infrarroja reflejada en las bandas alrededor de los 650 nm y 850 nm, usando métodos de segmentación de imágenes y clasificación de patrones. Igualmente se investigaron 2 tipos de aprendizaje automatizado como lo son las máquinas de soporte vectorial (SVM) y las redes neuronales convolucionales (CNN), que permitieron expandir los horizontes en materia de detección y monitoreo de anomalías presentes en las hojas de arroz y por consecuencia, en el cultivo, desarrollando así alternativas sólidas al momento de aplicarse a la agricultura de precisión mediante el uso de las herramientas necesarias para el testeado de estos algoritmos.

**Palabras Claves:** Agricultura de precisión, Procesamiento de Imágenes, Segmentación de Imágenes, NDVI, Matlab, Python, Machine Learning.

**ABSTRACT:**

In this project, a modification was developed and implemented in the multispectral image processing system in Matlab AppDesigner that allows the user to be alerted to the state of the crop in a segmented area through images from multispectral cameras or satellite images that calculate the NDVI value (Difference of Normalized Vegetation Index) based on measurements of the intensity of red and infrared light reflected in the bands around 650 nm and 850 nm, using image segmentation and pattern classification methods. Likewise, 2 Machine Learning methods (CNN and SVM) were investigated, which allowed expanding the horizons in terms of detection and monitoring of anomalies present in rice leaves and, consequently, in the crop, thus developing solid alternatives when applied to rice. precision agriculture through the use of the necessary tools for testing these algorithms.

**Keywords:** Precision Agriculture, Image Processing, Image Segmentation, NDVI, Matlab, Python, Machine Learning.



## **1. INTRODUCCIÓN:**

El problema de la protección eficaz de las enfermedades de las plantas está estrechamente relacionado con los problemas de la agricultura sostenible y el cambio climático. Los resultados de la investigación previa indican que el cambio climático puede alterar las etapas y las tasas de desarrollo de los patógenos; también puede modificar la resistencia del huésped, lo que conduce a cambios fisiológicos de las interacciones huésped-patógeno. La situación se complica aún más por el hecho de que, hoy en día, las enfermedades se transfieren globalmente con más facilidad que nunca.

Las nuevas enfermedades pueden aparecer en lugares en los que antes no estaban identificadas y, por lo tanto, donde no hay experiencia local para combatirlas. El uso inexperto de plaguicidas puede provocar el desarrollo de resistencia a largo plazo de los patógenos, reduciendo gravemente la capacidad de la planta de luchar contra la enfermedad. El diagnóstico oportuno y preciso de enfermedades de las plantas es uno de los pilares de la agricultura de precisión. Es crucial para evitar el desperdicio innecesario de recursos financieros y de otro tipo, logrando así una producción más sana, abordando el problema del desarrollo de la resistencia de los patógenos a largo plazo y mitigando los efectos negativos del cambio climático. En este entorno cambiante, la identificación adecuada y oportuna de la identificación de la enfermedad, incluida la prevención temprana, es más importante que nunca. Existen varias formas de detectar las patologías de las plantas. Algunas enfermedades no presentan síntomas visibles o el efecto se hace notar demasiado tarde para actuar, y en esas situaciones, es obligatorio un análisis sofisticado. Sin embargo, la mayoría de las enfermedades generan algún tipo de manifestación en el espectro visible, por lo que el examen a simple vista de un profesional capacitado es la principal técnica adoptada en la práctica para la detección de enfermedades de las plantas. Para lograr un diagnóstico preciso de las enfermedades de las plantas un fitopatólogo debe poseer una buena capacidad de observación para poder identificar los síntomas característicos.

Dicho lo anterior, se ve en la necesidad de indagar sobre nuevos métodos y técnicas de detección que permitan dar un diagnóstico acertado y oportuno sobre el estado de las plantas; en este trabajo se presentan herramientas vanguardistas que facilitan estas labores de observación, asimismo, aplicaciones de detección mediante diversos métodos: análisis mediante el valor del NDVI de imágenes multiespectrales, y también algoritmos y códigos de predicción y clasificación del estado de una planta. Se espera que dichas herramientas puedan ser implementadas a futuro de una manera más robusta y enfatizada en materia de prevención y detección de anomalías en los cultivos, facilitando así la labor en el control de calidad de las producciones del sembrado.

## 2. MARCO TEÓRICO/ESTADO DEL ARTE.

A continuación, se muestran las definiciones de los diferentes elementos teóricos que se usaron para llevar a cabo este proyecto. En la sección 2.1 se encuentran los antecedentes o estado del arte que se encontró antes de realizar el trabajo de investigación, la sección 2.2 y 2.3 se encuentran las bases teóricas y las partes más importantes de los códigos usados.

### 2.1 Marco de antecedentes.

En una revisión bibliográfica se encontró el artículo de investigación: “Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification” (Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016).). En este, se encuentra un nuevo enfoque para el desarrollo del modelo de reconocimiento de enfermedades de las plantas, basado en la imagen de la hoja clasificación, mediante el uso de redes convolucionales profundas. Esta novedosa forma de formación y la metodología utilizada facilitan una rápida y fácil implementación del sistema en la práctica. El modelo desarrollado es capaz de reconocer 13 tipos diferentes de enfermedades de las plantas a partir de hojas sanas, con la capacidad de distinguir las hojas de las plantas de su entorno. Para implementar este modelo de reconocimiento de enfermedades se inició con la recopilación de imágenes para crear una base de datos, evaluada por expertos agrícolas. Caffe, un *framework* de aprendizaje profundo desarrollado por Berkley Vision and Learning Centre, se utilizó para realizar el entrenamiento de la CNN. Los resultados experimentales en el modelo desarrollado lograron una precisión entre el 91% y el 98%, para pruebas de clases separadas, en promedio de 96,3%.

Se encontró otro artículo de investigación el cual tiene como título “Deep Learning for Plant Identification in Natural Environment” (Sun, Y., Liu, Y., Wang, G., & Zhang, H. (2017).). En este trabajo se presenta el primer conjunto de datos (*dataset*) de imágenes recopiladas por teléfono móvil en una escena natural, que contiene 10.000 imágenes de 100 especies de plantas ornamentales en el campus de la Universidad Forestal de Beijing. Un modelo de aprendizaje profundo de 26 capas que consta de 8 bloques de construcción residuales está diseñado para la clasificación de plantas a gran escala en un entorno natural. El modelo propuesto logra una tasa de reconocimiento del 91,78 % en el conjunto de datos BJFU100, demostrando que el aprendizaje profundo es una tecnología prometedora para la agricultura y silvicultura inteligente.

## 2.2 Marco teórico.

### 2.2.1. Índices de vegetación.

Índice de Vegetación Ratio (RVI):

Este índice puede usarse para estimar y monitorear la biomasa aérea, pues es muy efectivo para la estimación de la biomasa, especialmente en áreas con vegetación densa (Saavedra Mora, David & machado cuellar, Leidy & Méndez Pastrana, Diego & Murcia Torrejano, Valentín. (2020). Manual de interpretación y aplicación de imágenes multiespectrales en cultivos de importancia agrícola en el norte del Huila.). Fue propuesto por Pearson y Miller (1972), y es sensible a los efectos atmosféricos cuando la cubierta vegetal es inferior al 50 %, el cual se calcula según la siguiente ecuación (1):

$$RVI = \frac{\text{Banda Roja}}{\text{Infrarrojo Cercano}} \quad (1)$$

Índice de Vegetación de Diferencia (DVI):

Según Mehrotra y Srinivasan (2019) y Muñoz (2013), el DVI se desarrolló para distinguir entre el suelo y la vegetación. Este índice se expresa mediante una diferencia entre bandas rojas e infrarrojas cercanas (véase ecuación 2), es muy sensible a los cambios del suelo y tiene aplicabilidad en el monitoreo del ambiente ecológico de la vegetación. Fue propuesto por Richardson y Wiegang (1977).

$$DVI = \frac{\text{Infrarrojo Cercano}}{\text{Rojo}} \quad (2)$$

Índice de vegetación relacionado con NDVI (RG):

Con este índice pretendemos calcular un índice que excluya la banda del infrarrojo cercano y determinar la relación entre las bandas roja y verde con el índice de vegetación normalizado (NDVI); fue propuesto por García-Cervigón et al. (2015) (Saavedra Mora, David & machado cuellar, Leidy & Méndez Pastrana, Diego & Murcia Torrejano, Valentín. (2020). Manual de interpretación y aplicación de imágenes multiespectrales en cultivos de importancia agrícola en el norte del Huila). Su cálculo se realiza mediante la siguiente ecuación (3) (Mehrotra y Srinivasan, 2019):

$$RG = \frac{\text{Rojo}}{\text{Verde}} \quad (3)$$

Índice de vegetación de diferencia normalizado (NGRDI):

Este índice intenta predecir los valores de NDI utilizando solo bandas rojas y verdes. Por lo tanto, se puede aplicar a las imágenes recopiladas de cualquier

sensor RGB. Los valores también oscilan entre -1 y +1, como en el NDI, sugiriendo los más altos la presencia de plantas más saludables, lo cual fue propuesto por Gitelson et al. (2002). Sin embargo, no es tan preciso como el NDVI y necesita ser calibrado usando información terrestre para ser realmente útil (Mehrotra y Srinivasan, 2019) (figura 17). El cálculo del NGRDI implica el uso de una simple fórmula con dos bandas, la verde y la roja, y se determina mediante la siguiente ecuación (4):

$$NGRDI = \frac{Verde - Rojo}{Verde + Rojo} \quad (4)$$

Índice de vegetación de exceso de rojo (ExR):

Este índice se utiliza para determinar los píxeles, ya que es capaz de separar los píxeles de la planta en píxeles de fondo, y de esta manera segmentar las regiones foliares del fondo (Saavedra Mora, David & machado cuellar, Leidy & Méndez Pastrana, Diego & Murcia Torrejano, Valentín. (2020). Manual de interpretación y aplicación de imágenes multiespectrales en cultivos de importancia agrícola en el norte del Huila). Se calcula mediante la siguiente ecuación (5) (Mehrotra y Srinivasan, 2019):

$$ExR = 1,4 \cdot Rojo - Verde \quad (5)$$

Índice de vegetación normalizado diferencial (NDVI):

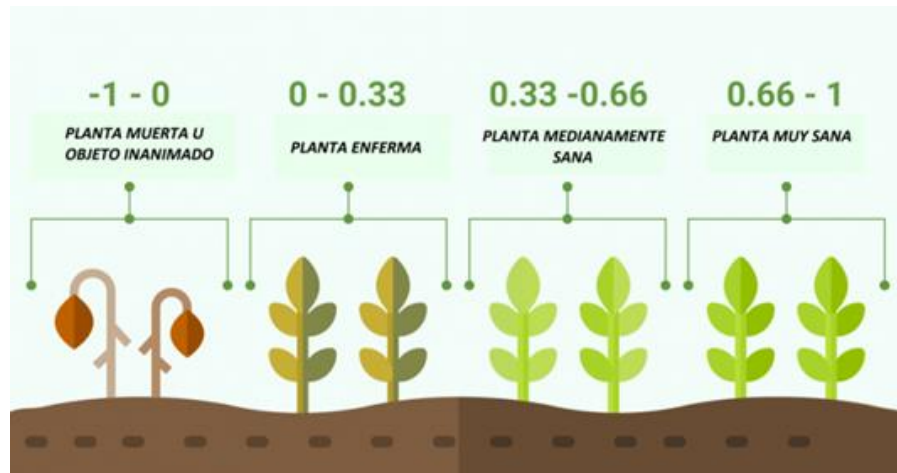
Dado que el índice se calcula a través de un procedimiento de normalización, el rango de valores NDVI está entre 0 y 1, y tiene una respuesta sensible a la vegetación verde, incluso en áreas cubiertas de vegetación baja. Este índice se usa a menudo en investigaciones relacionadas con evaluaciones de vegetación regionales y globales y se demostró que está relacionado no solo con la estructura del dosel y el LAI sino también con la fotosíntesis del dosel. Sin embargo, el NDVI es sensible a los efectos del brillo del suelo, el color del suelo, la atmósfera, la nube, la sombra de la nube y la sombra del dosel de la hoja, por lo que requiere calibración de detección remota. Asimismo, las plantas vivas bien nutridas absorben la luz roja y reflejan la luz del infrarrojo cercano. Sin embargo, también tiene en cuenta el hecho de que la vegetación estresada o muerta absorbe relativamente menos luz roja que la vegetación saludable, el suelo desnudo refleja tanto la luz roja como la del infrarrojo cercano de manera casi igual, y el agua abierta absorbe más infrarrojo que la luz roja. El NDVI se usa ampliamente debido a su simplicidad, y se han desarrollado varios índices para replicarlo o mejorarlo (Saavedra Mora, David & machado cuellar, Leidy & Méndez Pastrana, Diego & Murcia Torrejano, Valentín. (2020). Manual de interpretación y aplicación de imágenes multiespectrales en cultivos de importancia agrícola en el norte del Huila). Fue propuesto por Rouse et al. (1973). Se calcula mediante la siguiente ecuación (6) (Gracia et al., 2014; Mehrotra y Srinivasan, 2019):

$$NDVI = \frac{\text{Infrarrojo Cercano} - \text{Rojo}}{\text{Infrarrojo Cercano} + \text{Rojo}} \quad (6)$$

Los resultados del cálculo del NDVI varían de -1 a 1. Los valores negativos corresponden a áreas con superficies de agua, estructuras artificiales, rocas, nubes, nieve; el suelo desnudo generalmente cae dentro del rango de 0.1 a 0.2; y las plantas siempre tendrán valores positivos entre 0.2 y 1. El dosel de vegetación sano y denso debería estar por encima de 0.5, y la vegetación dispersa probablemente caerá dentro de 0.2 a 0.5 (Jaramillo-Véliz, 2021). Sin embargo, es solo una regla general y siempre debe tener en cuenta la temporada, el tipo de planta y las peculiaridades regionales para saber exactamente qué significan los valores de NDVI. (Toribio G., Grupo teledetección, 2021).

**Figura 1**

*Representación de los valores para los cálculos del NDVI.*



*Nota. Representación de los valores para los cálculos del NDVI, Toribio G., Grupo teledetección, 2021.*

En la mayoría de los casos, los valores de NDVI entre 0.2 y 0.4 corresponden a áreas con vegetación escasa; la vegetación moderada tiende a variar entre 0.4 y 0.6; cualquier cosa por encima de 0.6 indica la mayor densidad posible de hojas verdes.

### 2.2.2 Máquina de soporte vectorial.

Las máquinas de vectores de soporte o máquinas de vector soporte (del inglés support-vector machines, SVM) son un conjunto de algoritmos de aprendizaje supervisado desarrollados por Vladimir Vapnik y su equipo en los laboratorios de AT&T Bell.

Estos métodos están propiamente relacionados con problemas de clasificación y regresión. Dado un conjunto de ejemplos de entrenamiento (de muestras) podemos etiquetar las clases y entrenar una SVM para construir un modelo que prediga la clase de una nueva muestra. Intuitivamente, una SVM es un modelo que representa a los puntos de muestra en el espacio, separando las clases a 2 espacios lo más amplios posibles mediante un hiperplano de separación definido como el vector entre los 2 puntos, de las 2 clases, más cercanos al que se llama vector soporte (colaboradores de Wikipedia, 2022). Cuando las nuevas muestras se ponen en correspondencia con dicho modelo, en función de los espacios a los que pertenezcan, pueden ser clasificadas a una o la otra clase (Byvatov, E., & Schneider, G, 2003).

Más formalmente, una SVM construye un hiperplano o conjunto de hiperplanos en un espacio de dimensionalidad muy alta (o incluso infinita) que puede ser utilizado en problemas de clasificación o regresión. Una buena separación entre las clases permitirá una clasificación correcta (colaboradores de Wikipedia, 2022).

En un espacio  $p$ -dimensional, un hiperplano se define como un subespacio plano y afín de dimensiones  $p - 1$ . El término afín significa que el subespacio no tiene por qué pasar por el origen. En un espacio de dos dimensiones, el hiperplano es un subespacio de 1 dimensión, es decir, una recta. En un espacio tridimensional, un hiperplano es un subespacio de dos dimensiones, un plano convencional. Para dimensiones  $p > 3$  no es intuitivo visualizar un hiperplano, pero el concepto de subespacio con  $p - 1$  dimensiones se mantiene (SVM, n.d.).

La definición matemática de un hiperplano es bastante simple. En el caso de dos dimensiones, el hiperplano se describe acorde a la ecuación (7) de una recta (SVM, n.d.):

$$\beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 = 0 \quad (7)$$

Dados los parámetros  $\beta_0$ ,  $\beta_1$  y  $\beta_2$ , todos los pares de valores  $x = (x_1, x_2)$  (SVM, n.d.) para los que se cumple la igualdad son puntos del hiperplano. Esta ecuación (8) puede generalizarse para  $p$ -dimensiones (Hernández Guerrero, 2021):

$$\beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \dots + \beta_p \cdot x_p = 0 \quad (8)$$

y de igual manera, todos los puntos definidos por el vector  $x = (x_1, x_2, \dots, x_p)$  que cumplen la ecuación (9) y ecuación (10) pertenecen al hiperplano. Cuando  $x$  no satisface la ecuación (Hernández Guerrero, 2021):

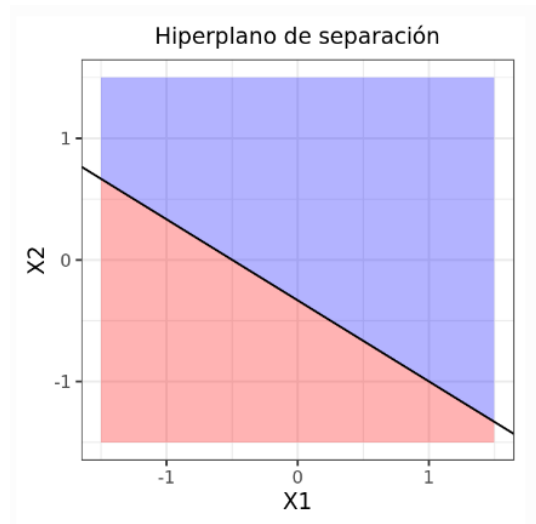
$$\beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \dots + \beta_p \cdot x_p < 0 \quad (9)$$

$$\beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \dots + \beta_p \cdot x_p > 0 \quad (10)$$

el punto  $x$  cae a un lado o al otro del hiperplano. Así pues, se puede entender que un hiperplano divide un espacio  $p$  – *dimensional* en dos mitades. Para saber en qué lado del hiperplano se encuentra un determinado punto  $x$ , solo hay que calcular el signo de la ecuación.

## Figura 2

*Hiperplano de separación de una SVM.*



La siguiente imagen muestra el hiperplano de un espacio bidimensional. La ecuación que describe el hiperplano (una recta) es  $1 + 2 \cdot x_1 + 3 \cdot x_2 = 0$ . La región azul representa el espacio en el que se encuentran todos los puntos para los que  $1 + 2 \cdot x_1 + 3 \cdot x_2 > 0$  y la región roja el de los puntos para los que  $1 + 2 \cdot x_1 + 3 \cdot x_2 < 0$ .

Los modelos basados en SVM están estrechamente relacionados con las redes neuronales. Usando una función kernel, resultan un método de entrenamiento alternativo para clasificadores polinomiales, funciones de base radial y perceptrón multicapa (Carral, 2021).

### **2.2.3 Redes neuronales convolucionales.**

Una red neuronal convolucional es un tipo de red neuronal artificial donde las neuronas artificiales, corresponden a campos receptivos de una manera muy similar a las neuronas en la corteza visual primaria (V1) de un cerebro biológico. Este tipo de red es una variación de un perceptrón multicapa, sin embargo, debido a que su aplicación es realizada en matrices bidimensionales, son muy efectivas para tareas de visión artificial, como en la clasificación y segmentación de imágenes, entre otras aplicaciones (Fernández Peña, 2019).

Las redes neuronales convolucionales consisten en múltiples capas de filtros convolucionales de una o más dimensiones. Después de cada capa, por lo general se añade una función para realizar un mapeo causal no-lineal (Fernández Peña, 2019).

Como redes de clasificación, al principio se encuentra la fase de extracción de características, compuesta de neuronas convolucionales y de reducción de muestreo (Fernández Peña, 2019). Al final de la red se encuentran neuronas de perceptron sencillas para realizar la clasificación final sobre las características extraídas. La fase de extracción de características se asemeja al proceso estimulante en las células de la corteza visual. Esta fase se compone de capas alternas de neuronas convolucionales y neuronas de reducción de muestreo (colaboradores de Wikipedia, 2022). Según progresan los datos a lo largo de esta fase, se disminuye su dimensionalidad, siendo las neuronas en capas lejanas mucho menos sensibles a perturbaciones en los datos de entrada, pero al mismo tiempo siendo estas activadas por características cada vez más complejas (Sun, Y., Liu, Y., Wang, G., & Zhang, H. 2017).

A continuación, se presenta un resumen de la arquitectura de una CNN:

- Entrada: Serán los píxeles de la imagen. Serán alto, ancho y profundidad será 1 sólo color o 3 para Red, Green, Blue, todo según la aplicación que se desee hacer.
- Capa De Convolución: procesará la salida de neuronas que están conectadas en “regiones locales” de entrada (es decir píxeles cercanos), calculando el producto escalar entre sus pesos (valor de píxel) y una pequeña región a la que están conectados en el volumen de entrada.



- CAPA RELU: aplicará la función de activación en los elementos de la matriz.
- POOL ó SUBSAMPLING: Hará una reducción en las dimensiones alto y ancho, pero se mantiene la profundidad.
- CAPA TRADICIONAL: red de neuronas feedforward que conectará con la última capa de subsampling y finalizará con la cantidad de neuronas que queremos clasificar.

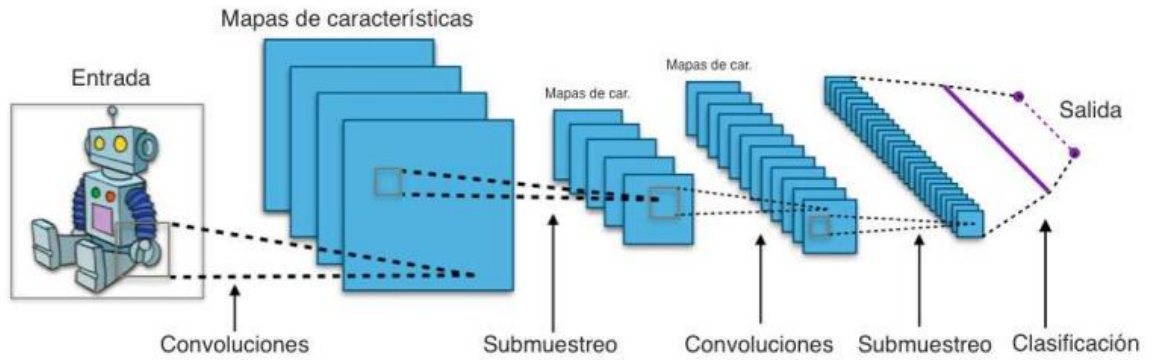
En síntesis, la arquitectura de una red neuronal convolucional empieza aplicando una capa convolucional y una capa de pooling, siguiendo este proceso repetidamente hasta obtener un conjunto de matrices, tal que al poner todos los elementos como un vector (proceso de flattening), tenga una cantidad computacionalmente viable de elementos como para ser introducido como entrada de una red neuronal (García Sánchez, n.d.).

Una red neuronal convolucional es un tipo de red multicapa que consta de diversas capas convolucionales y de pooling (submuestreo) alternadas, y al final tiene una serie de capas full-connected como una red perceptrón multicapa. La entrada de una red capa convolucional suele ser, generalmente, una imagen  $m \cdot m \cdot r$ , donde  $m$  es tanto la altura como el ancho de la imagen y  $r$  es el número de canales (como se verá después, en este proyecto se trabaja con escala de grises por lo que  $r = 1$ ). Las capas convolucionales tienen  $k$  filtros (o kernels) cuyas dimensiones son  $n \cdot n \cdot q$ , donde  $n$  y  $q$  son elegidas por el diseñador (generalmente  $q$  suele ser igual a  $r$ ). Cada filtro genera mediante convolución un mapa de rasgos o características de tamaño (Durán Suárez, 2017)  $(m - n + 1) \cdot (m - n + 1) \cdot p$ , siendo  $p$  el número de filtros que se desean usar. Después cada mapa es sub-muestreado en la capa de pooling con la operación “mean pooling” o “max pooling” sobre regiones contiguas de tamaño  $p \cdot p$  donde  $p$  puede tomar valores desde 2 para imágenes pequeñas hasta, comúnmente, no más de 5 para imágenes grandes (Ameijeiras Sánchez, David et al., 2020). Antes o después del submuestreo, se aplica una función de activación sigmoideal más un sesgo para cada mapa de rasgos (Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. 2016).

A continuación, se presenta una imagen que permite entender de mejor manera el concepto de una red neuronal convolucional:

Figura 3

Secuencia de una CNN.



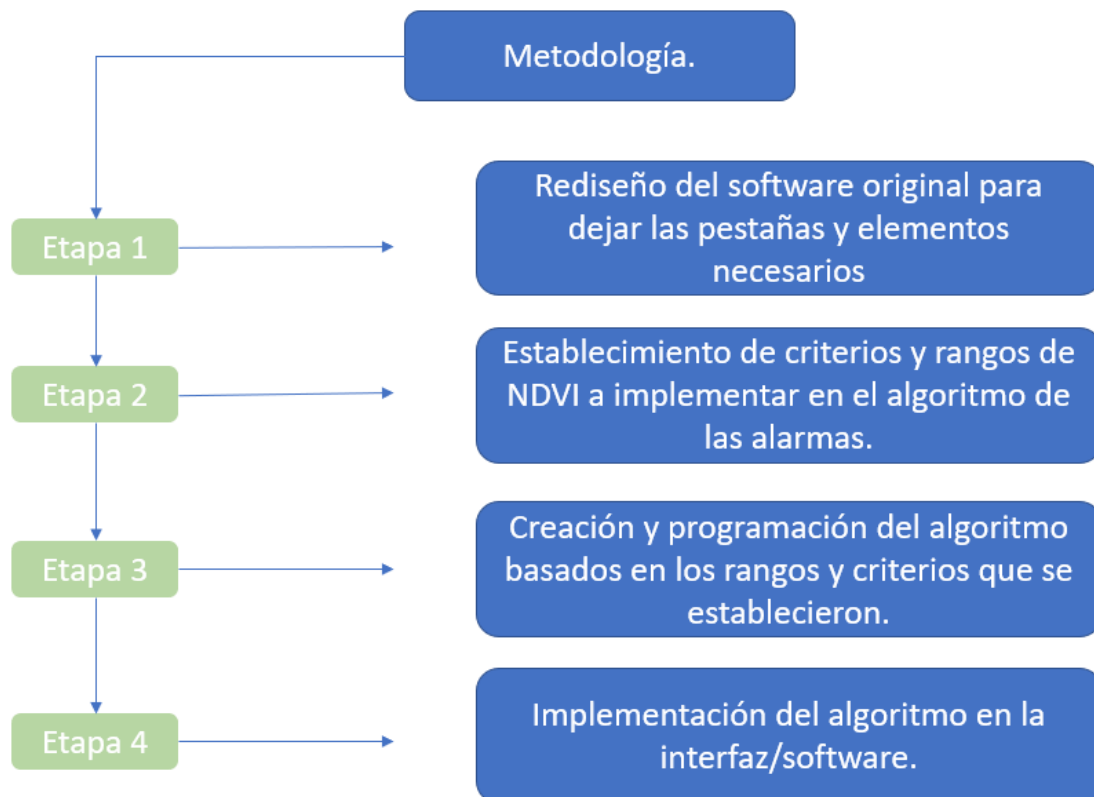
Nota. Secuencia de una CNN, de Pocho Costa, 2019.

### 3. MÉTODOS Y MATERIALES.

#### 3.1 Sistema de alarma.

Figura 4

Diagrama metodológico.



##### 3.1.1 NDVI.

A continuación, se detallan cada uno de los métodos que se usaron en cada etapa de desarrollo:

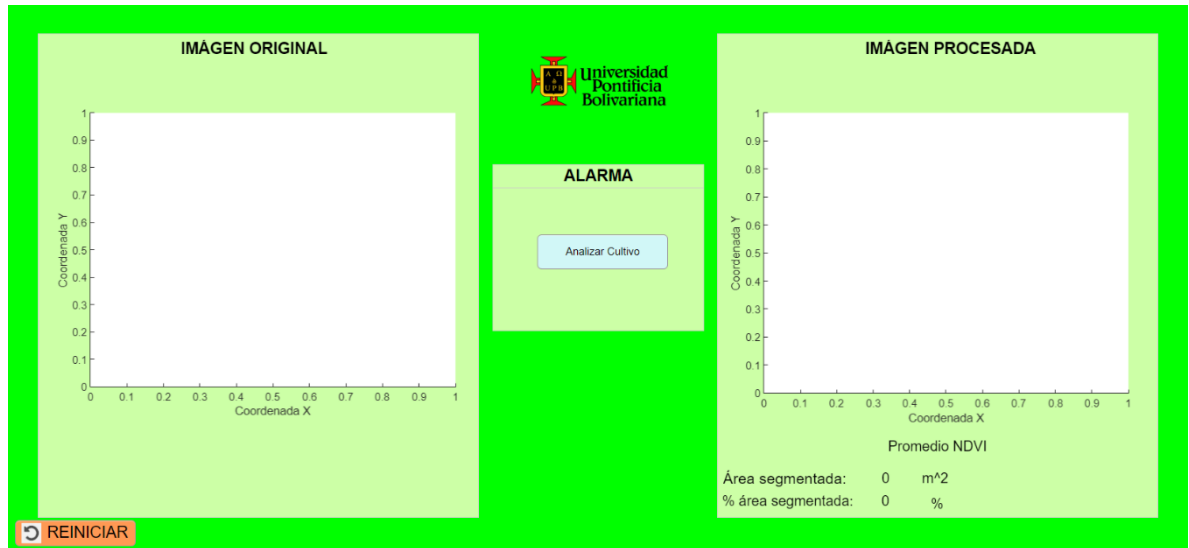
**Etapa 1:** Mediante el uso de la herramienta Matlab APP Designer, se accedió al código fuente de la interfaz perteneciente al software DRIoT- PROCESAMIENTO DE IMÁGENES PARA AGRICULTURA DE PRECISIÓN.

Para realizar las respectivas modificaciones como:

- Eliminación de pestañas no correspondientes al NDVI.
- Optimización y corrección de algunas líneas de código.
- En la figura 5 se observa la implementación de una nueva pestaña correspondiente al sistema de análisis y alarmas.

**Figura 5**

*Nueva Pestaña de alarma NDVI en la interfaz gráfica.*



**Etapa 2:** Se procede a hacer una investigación que permita determinar qué criterios se evalúan dentro de cada rango del valor del NDVI, para así tener un esquema de análisis y clasificación acertado según sea el valor del área segmentada. Los criterios más acertados fueron:

- 1 a 0: Planta muerta u objeto inanimado.
- 0,2 a 0,8: Vegetación fotosintéticamente activa.
- 0 a 0.33: planta enferma (dentro del rango de fotoactividad).
- 0.33 a 0.66: planta medianamente sana.

0.66 a 1: planta muy sana.

**Etap 3:** Se crea el código fuente del algoritmo por fuera de la interfaz para realizar pruebas independientes. Se agruparon los valores de los intervalos basados en 5 clases: planta muerta, planta viva, planta enferma, planta medianamente sana y planta muy sana.

**Figura 6**

*Algoritmo de alarma NDVI.*

```
my_icon = imread("pruebapg.jpg");
x = 0.5

if (-1 <= x ) && (x<=0);
    f = msgbox("La planta está muerta", "Error", "Error")
else
    f = msgbox("La planta está viva", "Verified", "custom", my_icon)

    if (0<x) && (x<=0.33);
        disp('La planta está enferma')
    elseif (0.33<x) && (x<=0.66);
        disp('La planta está medianamente sana')
    else
        disp('La planta está muy sana')
    end
end
```

**Etap 4:** Se implementa el código en la interfaz y se asigna la función a su respectivo botón.

**Figura 7**

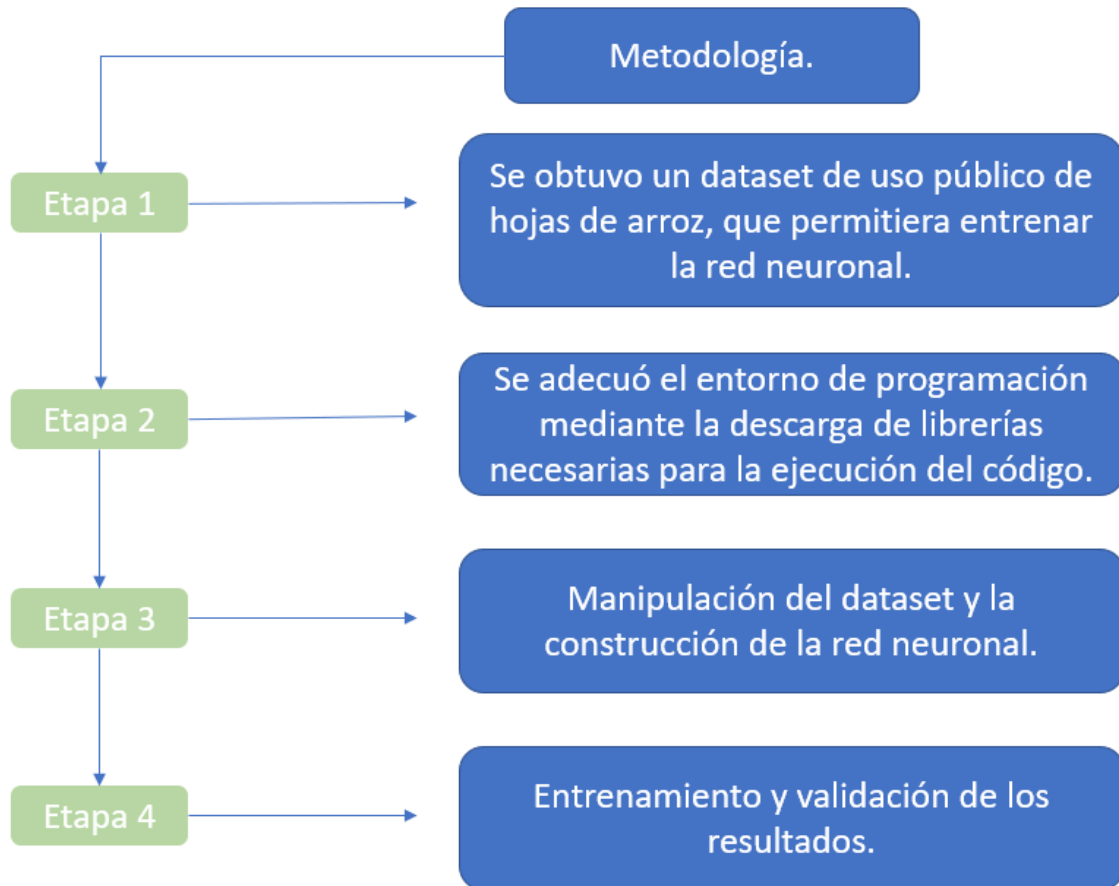
*Algoritmo de alarma NDVI dentro del código fuente de la interfaz.*

```
function AnalizarCultivoButtonPushed(app, event)
| global promedios R
if (-1 <= promedios(R) ) && (promedios(R) <=0)
    f = msgbox("El cultivo está muerto", "¡Alerta!", "error")
else
    if (0 < promedios(R)) && (promedios(R) <= 0.33)
        f = msgbox('El cultivo está enfermo', "¡Alerta!", "error")
    elseif (0.33 < promedios(R)) && (promedios(R) <= 0.66)
        f = msgbox('El cultivo está medianamente sano', "¡Cuidado!", "warn")
    else
        f = msgbox('El cultivo está muy sano');
    end
end
```

### 3.2 Algoritmos de aprendizaje (Deep Learning).

**Figura 8**

*Diagrama metodológico #2.*



#### 3.2.1 Red Neuronal Convolutacional.

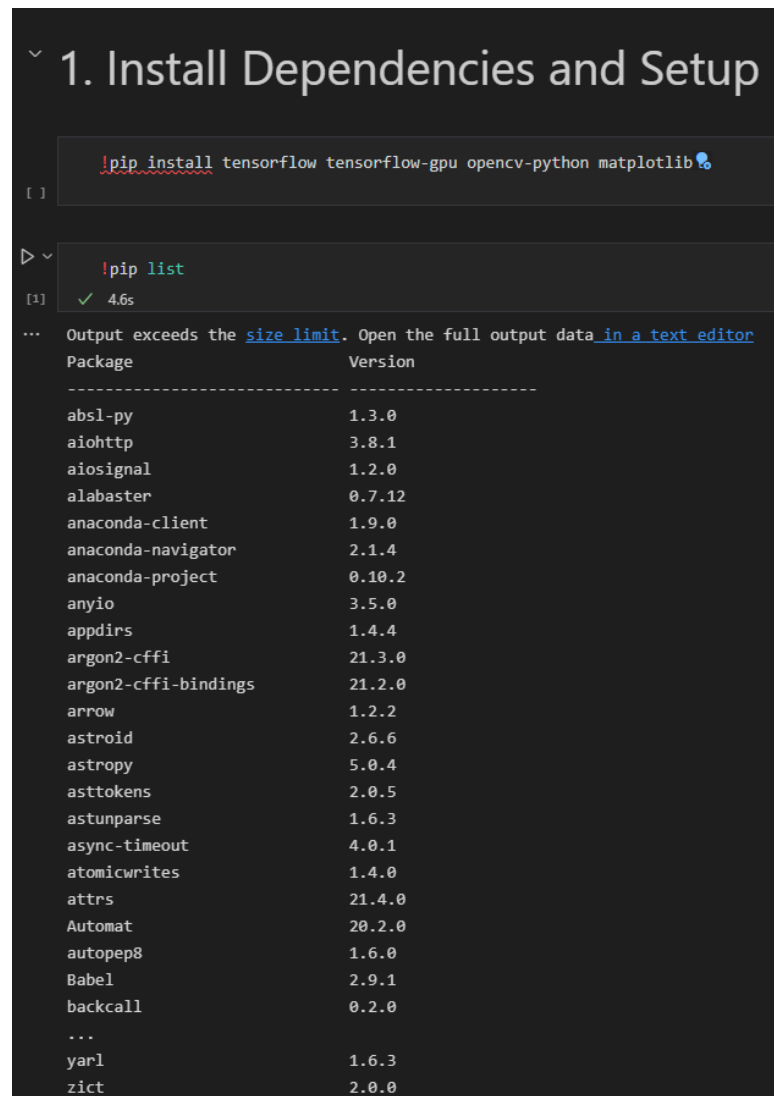
**Etapa 1:** Se accedió a la plataforma de Kaggle, que permite navegar y adquirir cientos de datasets de uso gratuito. En este caso se usó el dataset llamado Rice Leafs, el cual contiene datos de entrenamiento respecto a 4 condiciones (3 enfermedades y la condición sana) presentes en hojas de arroz.

**Etapa 2:** A continuación, se muestran las dependencias y librerías que permiten cargar los datos, visualizarlos y analizarlos. Así mismo las librerías que permiten

entrenar la red neuronal, para este trabajo se usó la biblioteca de TensorFlow, la cual es una biblioteca de código abierto para aprendizaje automático a través de un rango de tareas, y desarrollado por Google para satisfacer sus necesidades de sistemas capaces de construir y entrenar redes neuronales para detectar y descifrar patrones y correlaciones, análogos al aprendizaje y razonamiento usados por los humanos (Red neuronal artificial, the Glossary, s. f.).

### Figura 9

*Instalación y configuración de dependencias.*



```
1. Install Dependencies and Setup

!pip install tensorflow tensorflow-gpu opencv-python matplotlib

[ ]

!pip list

[1] ✓ 4.6s

... Output exceeds the size limit. Open the full output data in a text editor
Package                                Version
-----
absl-py                                 1.3.0
aiohttp                                  3.8.1
aiosignal                                1.2.0
alabaster                                 0.7.12
anaconda-client                          1.9.0
anaconda-navigator                       2.1.4
anaconda-project                         0.10.2
anyio                                     3.5.0
appdirs                                  1.4.4
argon2-cffi                              21.3.0
argon2-cffi-bindings                    21.2.0
arrow                                     1.2.2
astroid                                   2.6.6
astropy                                  5.0.4
asttokens                                 2.0.5
astunparse                               1.6.3
async-timeout                            4.0.1
atomicwrites                             1.4.0
attrs                                    21.4.0
Automat                                   20.2.0
autopep8                                  1.6.0
Babel                                     2.9.1
backcall                                  0.2.0
...
yarl                                     1.6.3
zict                                      2.0.0
```

**Etapa 3:** Se manipula el dataset de manera en que los datos de las condiciones se mezclan, para así obtener una salida binaria (basada en 2 criterios), ya sea que la hoja de la planta esté enferma de manera general, o se encuentre sana. Para la construcción de la red neuronal, los datos se re-escalan, se separan y se clasifican, de manera de que la red neuronal solo usa un porcentaje de los datos, en este caso 70% para el entrenamiento de la red neuronal, 20% para validación y un 10% para hacer las pruebas de las predicciones. A continuación, se muestran respectivamente las capturas de cada uno de estos procesos.

**Figura 10**

*Manipulación del dataset.*

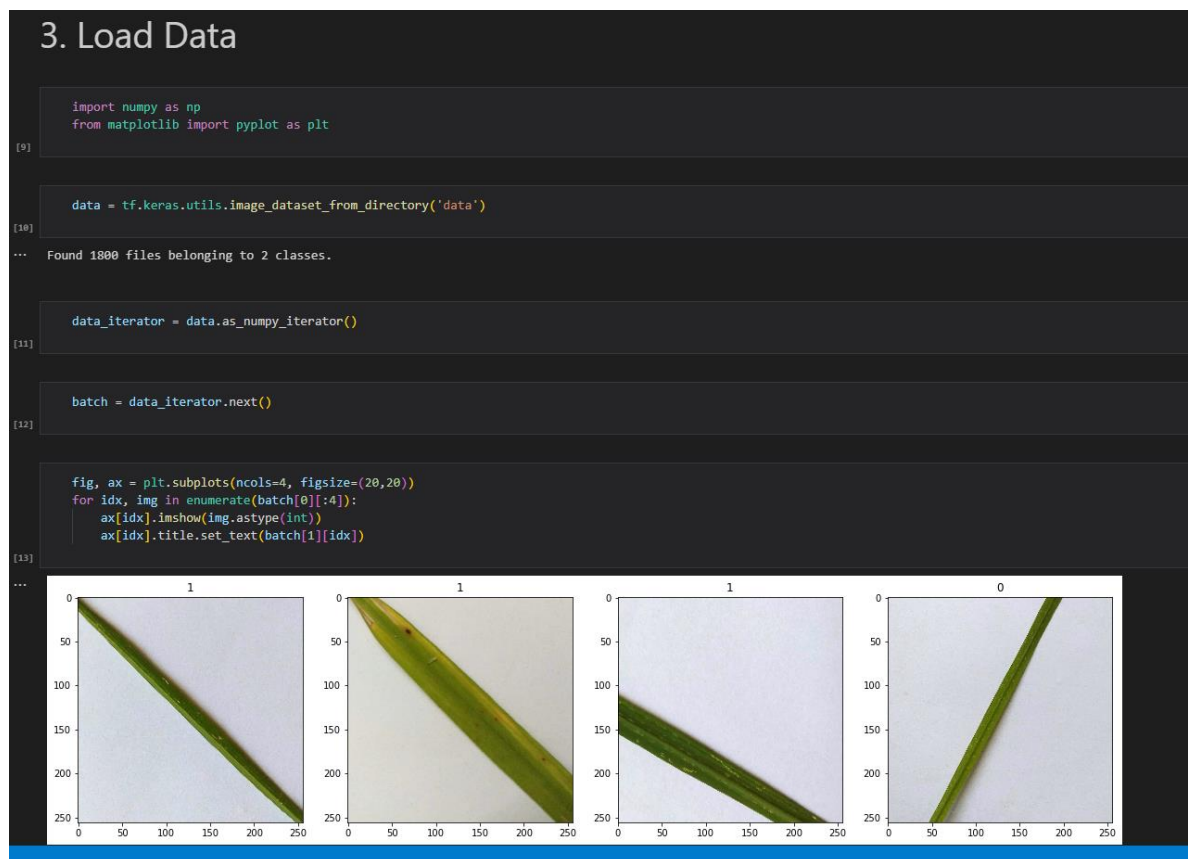




Figura 11

Re-escalamiento del conjunto de datos.

```
4. Scale Data

data = data.map(lambda x,y: (x/255, y))

[14]

data.as_numpy_iterator().next()

[15]

... Output exceeds the size limit. Open the full output data\_in\_a\_text\_editor
(array([[[[0.2969161 , 0.34005335, 0.07141622],
         [0.29721278, 0.34802622, 0.04076442],
         [0.2966745 , 0.3239787 , 0.02881547],
         ...,
         [0.8540775 , 0.8528456 , 0.90118617],
         [0.81336254, 0.81336254, 0.857327  ],
         [0.8414619 , 0.8414619 , 0.886826  ]],

        [[0.34421945, 0.38553378, 0.15261404],
         [0.28577366, 0.33675405, 0.06224425],
         [0.34186628, 0.36744452, 0.07361013],
         ...,
         [0.84728473, 0.84728473, 0.8933403  ],
         [0.83215743, 0.83215743, 0.87921625],
         [0.84506816, 0.8437975 , 0.89119023]],

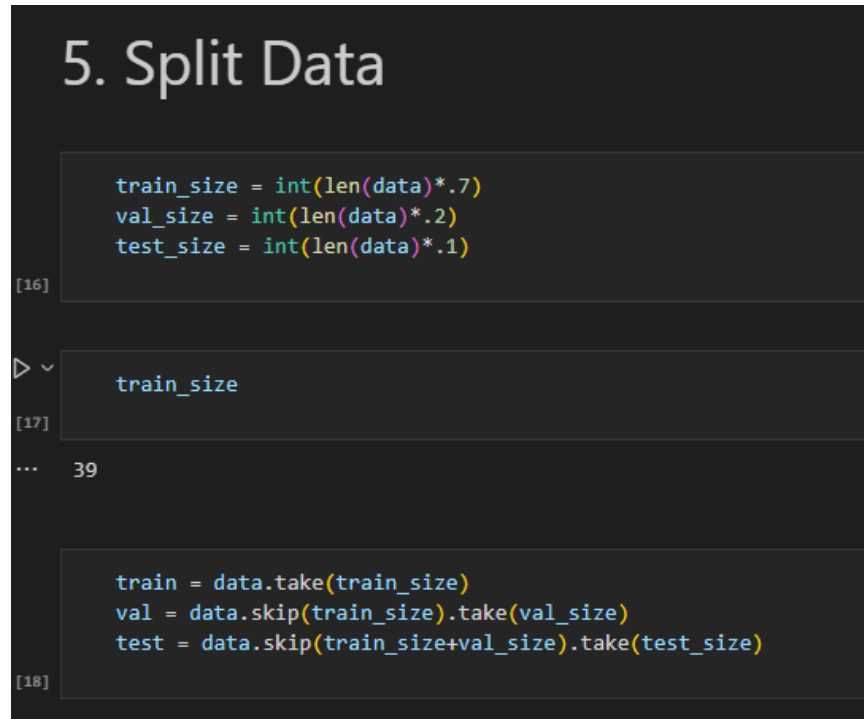
        [[0.7312216 , 0.7736785 , 0.6527724  ],
         [0.3171887 , 0.34701407, 0.14491542],
         [0.30726102, 0.33258134, 0.05807154],
         ...,
         [0.83491665, 0.83491665, 0.87943256],
         [0.8323486 , 0.8323486 , 0.8741071  ],
         [0.8103682 , 0.8103682 , 0.8521268  ]],

        ...,

        [0.8489641 , 0.8450425 , 0.8999445  ],
         [0.85475093, 0.85082936, 0.9057313  ],
         [0.84338164, 0.8394601 , 0.89436203]]], dtype=float32),
```

Figura 12

Separación de datos.



The image shows a Jupyter Notebook interface with a dark background. At the top, the title "5. Split Data" is displayed in white. Below the title, there are three code cells. The first cell, labeled [16], contains the following Python code: `train_size = int(len(data)*.7)`, `val_size = int(len(data)*.2)`, and `test_size = int(len(data)*.1)`. The second cell, labeled [17], contains the code `train_size` and shows a variable inspection window with the value 39. The third cell, labeled [18], contains the code `train = data.take(train_size)`, `val = data.skip(train_size).take(val_size)`, and `test = data.skip(train_size+val_size).take(test_size)`.

**Etapa 4:** En esta etapa se observan y validan los resultados que arroja la red neuronal después de su entrenamiento, también se adjunta el algoritmo que permite evaluar los resultados; cabe mencionar que para las pruebas se usaron 20 iteraciones.

Figura 13

Evaluación y entrenamiento de la red neuronal.

```
9. Evaluate

from tensorflow.keras.metrics import Precision, Recall, BinaryAccuracy

pre = Precision()
re = Recall()
acc = BinaryAccuracy()

for batch in test.as_numpy_iterator():
    X, y = batch
    yhat = model.predict(X)
    pre.update_state(y, yhat)
    re.update_state(y, yhat)
    acc.update_state(y, yhat)

... 1/1 [=====] - 1s 697ms/step
1/1 [=====] - 0s 337ms/step
1/1 [=====] - 0s 337ms/step
1/1 [=====] - 0s 430ms/step
1/1 [=====] - 0s 338ms/step

print(pre.result(), re.result(), acc.result())

... tf.Tensor(0.8243243, shape=(), dtype=float32) tf.Tensor(0.6777777, shape=(), dtype=float32) tf.Tensor(0.7375, shape=(), dtype=float32)
```

Figura 14

Testeo y validación de resultados.

```
10. Test

import cv2

img = cv2.imread('D:\Descargas\Dataset\ImageClassification-main\ImageClassification-main\Pruebas\sanas\IMG_20190419_115220.jpg')
plt.imshow(img)
plt.show()

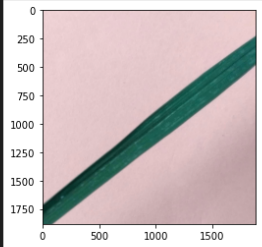
...

```

Figura 15

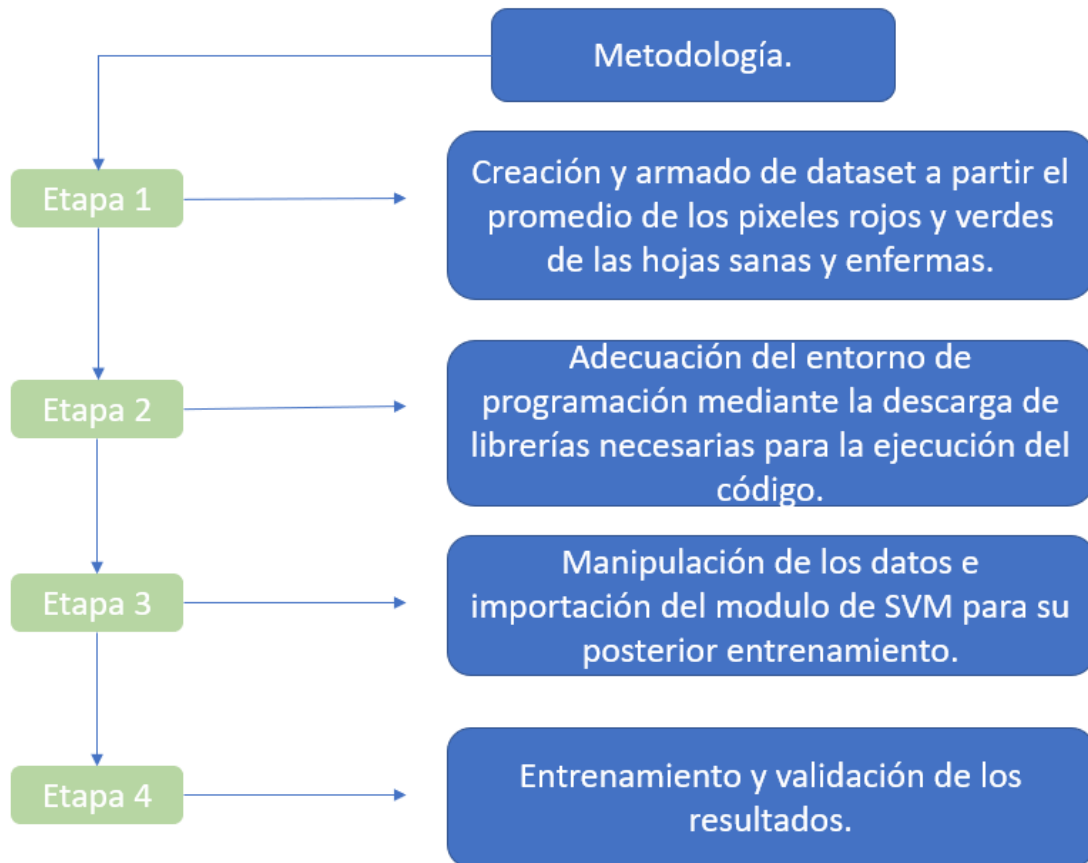
Testeo y validación de resultados



### 3.2.2 Máquina de soporte vectorial.

**Figura 16**

*Diagrama metodológico #3.*



**Etapa 1:** Se desarrolló un programa que permitiera leer las imágenes de hojas de arroz contenidas en el dataset y mediante este programa, separar las imágenes en sus 3 bandas de colores, luego de hacer esta separación, se implementa la función de promedio para obtener los valores de la banda roja y verde. Cabe destacar que las imágenes estaban previamente clasificadas según el estado de la planta (sana o enferma), en total se usaron 306 imágenes. Consiguientemente, una vez obtenido el valor de promedio de los pixeles rojos y verdes se agregó una columna al dataset llamada "estado planta" que se le asigna el valor de 0 o 1 según a que clase pertenezca el dato, una vez organizados los datos de las 3 columnas, se crea un archivo .csv, el cual será importado en la IDE de Jupyter.

**Etapa 2:** Se utilizaron las siguientes librerías para el correcto funcionamiento del algoritmo:

“matplotlib.pyplot”: librería encargada de realizar las gráficas.

“pandas”: librería con la que se pueden analizar y manipular datos, en este caso se usa para leer el dataset.

“seaborn”: librería encargada de la visualización de los datos.

“numpy”: librería encargada de la realización de operaciones matemáticas.

“matplotlib.colors”: librería encargada de asignar los colores correspondientes a la zona de clasificación de la svm y a los vectores de soporte.

“chart\_studio”: librería que permite visualizar los datos e interactuar con las gráficas.

“Sklearn”: librería que permite importar los modelos de SVM.

**Etapa 3:** Se hace uso de la librería panda para leer el dataset, luego se separan los valores de entrada y salida en 2 variables, denominadas “X” y “Y”, y se observan que datos corresponden a cada clase. Una vez hecho esto, se importa el modelo de la SVM y se le asigna el 20% de los datos para las pruebas o testeo y 80% para el entrenamiento.

**Etapa 4:** Se observan los resultados mediante la gráfica de clasificación y se determina el valor de la precisión de esta misma.

#### 4. RESULTADOS Y DISCUSIÓN.

En este trabajo de investigación se desarrollaron, implementaron y ejecutaron códigos que permiten la detección y reconocimiento de anomalías en cultivos y hojas de arroz. Primeramente, se obtuvo una modificación del software capaz de alertar al usuario si hay anomalías al analizar el área segmentada por NDVI de las imágenes multiespectrales, el software es capaz de segmentar la imagen en 68 áreas, una vez se localice el área de interés y su valor de NDVI, el usuario puede acceder a la pestaña de alarmas NDVI y hacer click en el botón analizar cultivo, consecuentemente, la aplicación mostrará mediante una ventana emergente el estado en el que se encuentra la zona que se esté analizando. La interfaz realiza la detección y alarma mediante los rangos NDVI que se muestran en la figura 17 y 18.

**Figura 17**

*Segmentación y selección de áreas en la interfaz NDVI.*

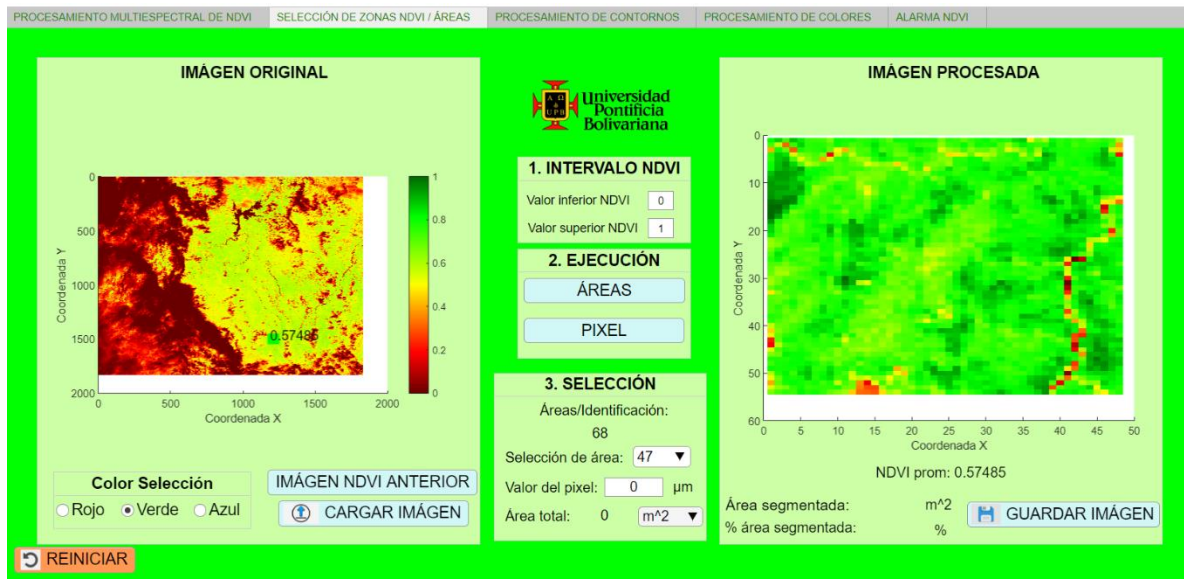
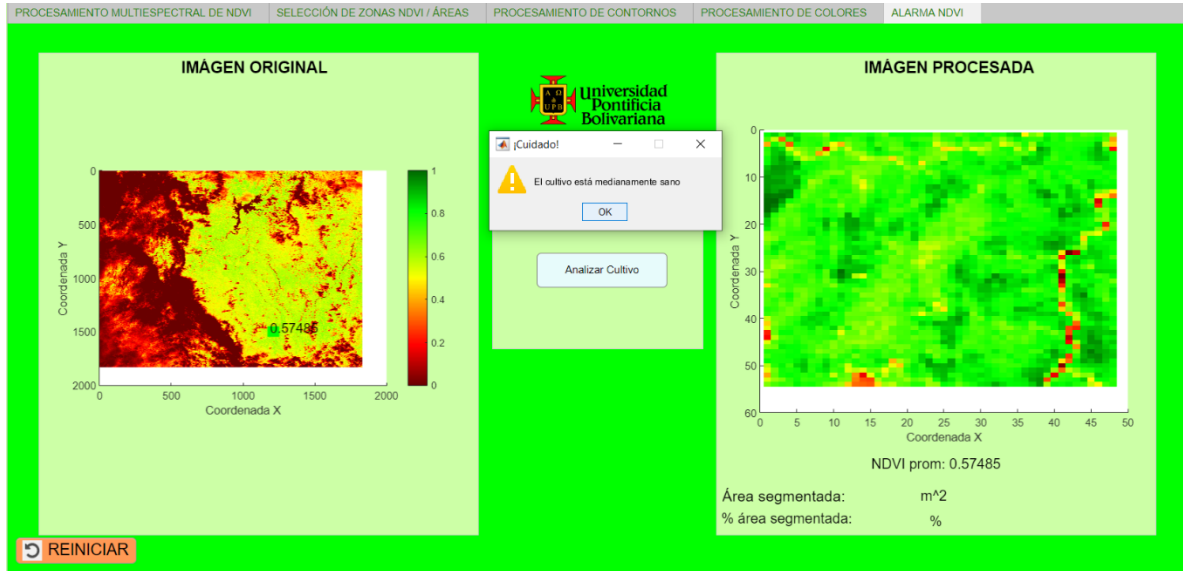


Figura 18

Análisis del área segmentada y su respectiva alarma que define el estado.



La interfaz de alarmas NDVI permite evaluar de manera general un área segmentada de una imagen multiespectral proveniente de algún cultivo, lo que supone un gran avance para diagnosticar y prevenir que zonas se encuentran en un potencial estado que pueda comprometer la productividad esperada del cultivo. En esa misma línea, se testearon y ejecutaron en Python 2 algoritmos de aprendizaje automatizado que permitieron ir un paso más allá en lo que a detección de anomalías se refiere. Principalmente, se investigó y testeó un algoritmo de red neuronal convolucional sencilla, en el cual, mediante el uso de un dataset de imágenes de hojas de arroz, se pueden identificar patrones que permitan a la red neuronal determinar y clasificar si la hoja se encuentra enferma o sana, la precisión de este algoritmo depende de las iteraciones que se le pida a la red neuronal realizar, para efectos de este trabajo, se usaron 20 iteraciones ya que estas consumen una capacidad de procesamiento aceptable arrojando una precisión del 92%, como se puede apreciar en la figura 19. Esto supone un indicador relevante, ya que según sea el estado de la hoja se puede determinar el estado del cultivo y proceder a hacer una revisión en campo sobre el estado real y actual del sembrado. De manera alternativa, también se tuvo como materia de estudio y desarrollo, la implementación de una máquina de soporte vectorial binaria que permitiera mediante el análisis de color de los píxeles rojos y verdes de las imágenes de hojas de arroz clasificar el estado de la planta ya sea que estuviera enferma o sana, la manipulación de este código, permitió asimismo, crear un dataset, ya que naturalmente no habían datos de entradas para montar



sobre la máquina de soporte vectorial; una vez manipulado el dataset, se pudo testear el código para así analizar los resultados y determinar que método de Machine Learning es más preciso y viable para caracterizar patrones que representen el estado de la hoja/planta, en el caso de la SVM también se obtuvo un resultado muy favorable en la precisión al momento de clasificar, siendo este valor de 93.5%, ligeramente más preciso que la red neuronal.

### Figura 19

*Evaluación de la precisión de la CNN.*



```
9. Evaluate

[312] from tensorflow.keras.metrics import Precision, Recall, BinaryAccuracy Python

[313] pre = Precision()
      re = Recall()
      acc = BinaryAccuracy() Python

[314] for batch in test.as_numpy_iterator():
      X, y = batch
      yhat = model.predict(X)
      pre.update_state(y, yhat)
      re.update_state(y, yhat)
      acc.update_state(y, yhat) Python

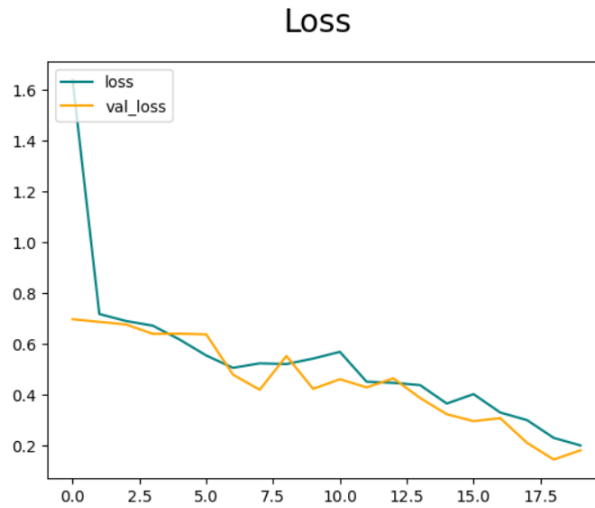
... 1/1 [=====] - 0s 417ms/step

[315] print(pre.result(), re.result(), acc.result()) Python

... tf.Tensor(0.9285714, shape=(), dtype=float32) tf.Tensor(0.9285714, shape=(),
dtype=float32) tf.Tensor(0.9375, shape=(), dtype=float32)
```

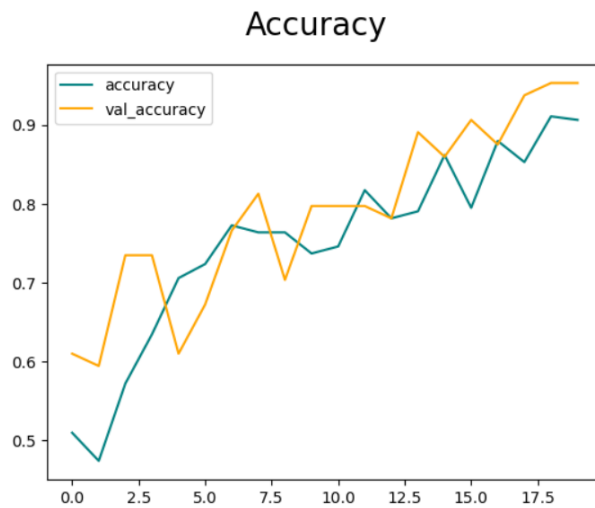
**Figura 20**

*Perdidas de la CNN.*



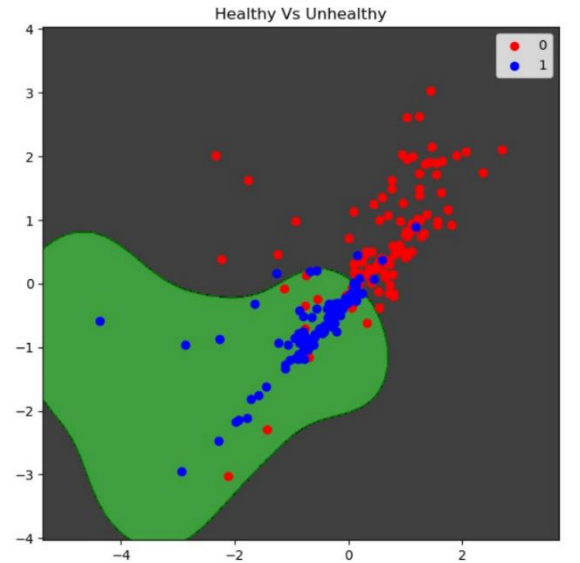
**Figura 21**

*Precisión de la CNN.*



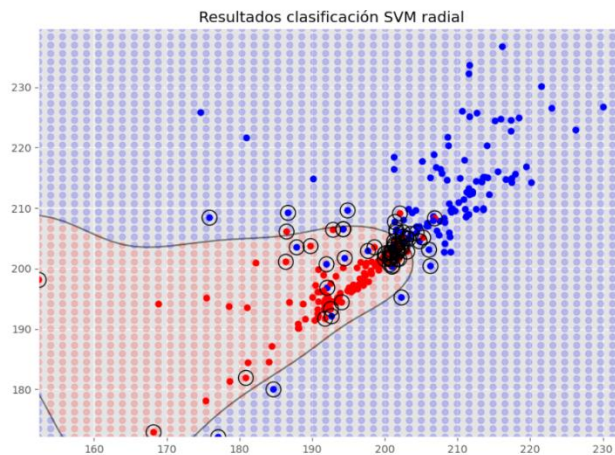
**Figura 22**

*Clasificación de la SVM.*



**Figura 23**

*Clasificación de la SVM con Hiperparámetros.*



En la figura 23, se puede observar el resultado de la clasificación, pero esta vez con los mejores hiperparámetros encerrados, esto ayuda a que cuando se

tengan nuevos datos para clasificar tomen como referencia a los datos que son hiperparámetros, ya que estos son los más significativos al momento en que la SVM ejecuta su proceso de clasificación, es por esto que es de suma relevancia saber cuáles son los hiperparámetros y sus valores, ya que datos cercanos a estos serán clasificados de una manera más precisa y eficiente. En la figura 22 se aprecia la clasificación básica de la SVM. Las figuras 20 y 21 corresponden a los resultados de las pérdidas como de la precisión de la red neuronal convolucional. Haciendo un análisis y relación entre ambas, se puede observar que, la relación pérdida/precisión es favorable arrojando una precisión teórica del 91%.

## **5. CONCLUSIONES Y RECOMENDACIONES.**

Hay muchos métodos en el proceso de detección y clasificación de enfermedades de las plantas, pero, aun así, este campo de la investigación es escaso. Además, todavía no hay soluciones comerciales en el mercado, excepto las que se ocupan del reconocimiento de especies vegetales basado en las imágenes de las hojas. En este trabajo, se exploró un nuevo enfoque del uso de los algoritmos del modelo de aprendizaje automático Deep Learning para clasificar y detectar el estado de las plantas a partir de imágenes de las hojas. El modelo testeado fue capaz de detectar la presencia de hojas y distinguir entre hojas sanas y 3 enfermedades diferentes, que pueden ser diagnosticadas visualmente. Se describió el procedimiento completo, respectivamente, desde la recolección de las imágenes utilizadas para el entrenamiento y la validación hasta el preprocesamiento y aumento de la imagen y, finalmente el procedimiento de entrenamiento de la CNN profunda y la puesta a punto. Se realizaron diferentes pruebas para comprobar el rendimiento del modelo recién creado. Se implementó una base de datos de imágenes de enfermedades de plantas, que contenía más de 3.000 imágenes originales tomadas de las fuentes disponibles en Internet y ampliada a más de 5.000 utilizando las transformaciones adecuadas. Los resultados experimentales alcanzaron una precisión de entre el 88% y el 91%, para las pruebas de clases separadas de clase. La precisión global final del modelo entrenado fue del 93%, se recomienda para próximas investigaciones (si se cuenta con una capacidad de procesamiento más alta), aumentar el número de iteraciones para obtener aún más precisión. El ajuste fino no ha mostrado cambios significativos en la precisión global, pero el proceso de aumento de números de iteraciones tuvo mayor influencia para lograr resultados respetables. Como el método presentado no ha sido explotado, por lo que sabemos, en el campo del reconocimiento de enfermedades de las plantas, no se han comparado los resultados con otros relacionados, utilizando la misma técnica. En lo que respecta al análisis mediante NDVI, esto supone una alternativa diferente en el desarrollo de herramientas de detección y clasificación, ya que permite mediante valores cuantitativos determinar el posible estado general de la zona que se esté estudiando y hacer una valoración cualitativa para hacer un estudio que permita determinar si es necesaria la intervención humana dentro del área que se esté analizando.

Una posible ampliación de este trabajo de grado consistirá en reunir imágenes para enriquecer la base de datos y mejorar la precisión del modelo utilizando diferentes técnicas de ajuste y aumento. El objetivo principal del trabajo futuro puede ser desarrollar un sistema completo que conste de una interfaz robusta que contenga un modelo de aprendizaje automatizado entrenado y una aplicación para dispositivos móviles inteligentes o computadoras con características como la visualización de enfermedades reconocidas en frutas,

hortalizas y otras plantas, basándose en las imágenes de las hojas capturadas por la cámara de cualquier dispositivo. Esta aplicación servirá de ayuda a los agricultores (independientemente del nivel de experiencia), permitiendo un reconocimiento rápido y eficaz de las enfermedades de las plantas y facilitar el proceso de toma de decisiones a la hora de tomar decisiones sobre el uso de plaguicidas químicos. Además, el trabajo futuro consistirá en difundir el uso del modelo, entrenándolo para el reconocimiento de enfermedades vegetales en zonas más amplias, combinando fotos aéreas de huertos y viñedos captadas por drones y redes neuronales de convolución para la detección de objetos. Al ampliar esta investigación se espera conseguir un valioso impacto en el desarrollo sostenible, afectando a la calidad de los cultivos para las generaciones futuras.

## 6. BIBLIOGRAFÍA.

- Barnaby, J. Y., Huggins, T. D., Lee, H., McClung, A. M., Pinson, S. R. M., Oh, M., Baughan, G. R., Tarpley, L., Lee, K., Kim, M. S., & Edwards, J. D. (2020). Vis/NIR hyperspectral imaging distinguishes sub-population, production environment, and physicochemical grain properties in rice. *Scientific Reports*, 10(1). <https://doi.org/10.1038/s41598-020-65999-7>
- Basterra, I. (2011). Cátedra de Fotointerpretación, Departamento de Geociencias Facultad de Ingeniería - UNNE. 33 pág.
- Cen, H., Wan, L., Zhu, J., Li, Y., Li, X., Zhu, Y., Weng, H., Wu, W., Yin, W., Xu, C., Bao, Y., Feng, L., Shou, J., & He, Y. (2019). Dynamic monitoring of biomass of rice under different nitrogen treatments using a lightweight UAV with dual image-frame snapshot cameras. *Plant Methods*, 15(1), 32. <https://doi.org/10.1186/s13007-019-0418-8>
- Falco, N., Wainwright, H. M., Dafflon, B., Ulrich, C., Soom, F., Peterson, J. E., Brown, J. B., Schaettle, K. B., Williamson, M., Cothren, J. D., Ham, R. G., McEntire, J. A., & Hubbard, S. S. (2021). Influence of soil heterogeneity on soybean plant development and crop yield evaluated using time-series of UAV and ground-based geophysical imagery. *Scientific Reports*, 11(1). <https://doi.org/10.1038/s41598-021-86480-z>
- Franz (2018). "Las firmas espectrales, conoce sus bandas espectrales en la Teledetección". GISGeography. Disponible en: <https://acolita.com/las-firmas-espectrales-conoce-sus-bandas-espectrales-en-la-teledeteccion/>
- García J., S. Andres., & Martínez M, L. Joel. (2010). *metodo para identificacion de cultivos de arroz con base en imagenes de satelite*.
- García, C. E., Herrera, F. A., & Erazo Mesa, E. (2014). *Metodología básica para la generación de índices de vegetación mediante imágenes multiespectrales aerotransportadas aplicada en cultivos de caña de azúcar*.
- García-Cervigón, D., José Junio, J., Rodríguez, G., & del Pilar Gutierrez Puebla, M. (2015). *UNIVERSIDAD COMPLUTENSE DE MADRID FACULTAD DE GEOGRAFÍA E HISTORIA TRABAJO FIN DE MÁSTER Estudio de Índices de vegetación a partir de imágenes aéreas tomadas desde UAS/RPAS y aplicaciones de estos a la agricultura de precisión*.
- García-Cervigón, D.; José, J.; Rodríguez, G.; y Del Pilar Gutiérrez Puebla, M. (2014). "Estudio de índices de vegetación a partir de imágenes aéreas tomadas desde UAS/RPAS y aplicaciones de estos a la agricultura de precisión". Disponible en: [http://eprints.ucm.es/31423/1/TFM\\_Juan\\_Diaz\\_Cervignon.pdf](http://eprints.ucm.es/31423/1/TFM_Juan_Diaz_Cervignon.pdf).
- González, F. A., Gómez, J. J., & Amaya, D. F. (2017). *Procesamiento de imágenes multiespectrales en cultivos de café y cacao*.
- Guadalupe Sánchez-Méndez, A., & Pedro Arguijo-Hernández, S. (2018). Análisis de imágenes multiespectrales para la detección de cultivos y detección de plagas y enfermedades en la producción de café. In *Research in Computing Science* (Vol. 147, Issue 7).
- Hernández Molina, D. D., & Gulfo Galaraga, J. M. (2020). *PROCESAMIENTO DE IMÁGENES MULTIESPECTRALES ADQUIRIDAS CON DRONES PARA LA CARACTERIZACIÓN DE SISTEMAS AGROAMBIENTALES*.
- Hossen, M. A., Diwakar, P. K., & Ragi, S. (2021). Total nitrogen estimation in agricultural soils via aerial multispectral imaging and LIBS. *Scientific Reports*, 11(1). <https://doi.org/10.1038/s41598-021-90624-6>  
<https://doi.org/10.1007/s11119-012-9274-5>
- Inoue, Y., Sakaiya, E., Zhu, Y., & Takahashi, W. (2012). Diagnostic mapping of canopy nitrogen content in rice based on hyperspectral measurements. *Remote Sensing of Environment*, 126, 210–221. <https://doi.org/10.1016/j.rse.2012.08.026>
- Instituto Geográfico Nacional de España. (2016). *Teledetección*. <http://www.ign.es>
- Jay, S., Baret, F., Dutartre, D., Malatesta, G., Héno, S., Comar, A., Weiss, M., & Maupas, F. (2019). Exploiting the centimeter resolution of UAV multispectral imagery to improve remote-

- sensing estimates of canopy structure and biochemistry in sugar beet crops. *Remote Sensing of Environment*, 231, 110898. <https://doi.org/https://doi.org/10.1016/j.rse.2018.09.011>
- Kerr, J., & Ostrovsky, M. (2003). Kerr JT and Ostrovsky M. From space to species: ecological applications for remote sensing. *Trends Ecol Evol. Trends in Ecology & Evolution*, 18, 299–305. [https://doi.org/10.1016/S0169-5347\(03\)00071-5](https://doi.org/10.1016/S0169-5347(03)00071-5)
- Khaki, S., Pham, H., & Wang, L. (2021). Simultaneous corn and soybean yield prediction from remote sensing data using deep transfer learning. *Scientific Reports*, 11(1). <https://doi.org/10.1038/s41598-021-89779-z>
- Kharuf-Gutierrez, S., Hernández-Santana, L., Orozco-Morales, R., de la Aday Díaz, O. C., & Delgado Mora, I. (2018). Análisis de imágenes multiespectrales adquiridas con vehículos aéreos no tripulados. In *RIELAC* (Issue 2).
- Lassalle, G., Fabre, S., Credoza, A., Hédacq, R., Dubucq, D., & Elger, A. (2021). Mapping leaf metal content over industrial brownfields using airborne hyperspectral imaging and optimized vegetation indices. *Scientific Reports*, 11(1). <https://doi.org/10.1038/s41598-020-79439-z>
- Maimaitijiang, M., Sagan, V., Sidike, P., Hartling, S., Esposito, F., & Fritsch, F. B. (2020). Soybean yield prediction from UAV using multimodal data fusion and deep learning. *Remote Sensing of Environment*, 237, 111599. <https://doi.org/https://doi.org/10.1016/j.rse.2019.111599>
- Martí, S. P. de, & Agr, I. (n.d.). *AGRICULTURA DE PRECISIÓN Y PROTECCIÓN DE CULTIVOS*. *Precision crop protection*. <https://sentinel.esa.int/web/sentinel/missions/>
- Mehrotra, N. y Srinivasan, S. (2019). "Analysing Drone and Satellite Imagery using Vegetation Indices".
- Morel, J., Jay, S., Féret, J. B., Bakache, A., Bendoula, R., Carreel, F., & Gorretta, N. (2018). Exploring the potential of PROCOSINE and close-range hyperspectral imaging to study the effects of fungal diseases on leaf physiology. *Scientific Reports*, 8(1). <https://doi.org/10.1038/s41598-018-34429-0>
- Moreno-García, B., Guillén, M., Casterad, M. A., & Quílez, D. (2013). *USO DE IMÁGENES AÉREAS MULTIESPECTRALES PARA ESTIMACIÓN DEL RENDIMIENTO EN CULTIVO DE ARROZ*.
- Mu, Y., Fujii, Y., Takata, D., Zheng, B., Noshita, K., Honda, K., Ninomiya, S., & Guo, W. (2018). Characterization of peach tree crown by using high-resolution images from an unmanned aerial vehicle. *Horticulture Research*, 5(1). <https://doi.org/10.1038/s41438-018-0097-z>
- Muñoz Aguayo, P. (2013). Apuntes de teledetección: índices de vegetación. Centro de Información de Recursos Naturales. 15 pág.
- Nehoshtan, Y., Carmon, E., Yaniv, O., Ayal, S., & Rotem, O. (2021). Robust seed germination prediction using deep learning and RGB image data. *Scientific Reports*, 11(1). <https://doi.org/10.1038/s41598-021-01712-6>
- Pan, W. J., Wang, X., Deng, Y. R., Li, J. H., Chen, W., Chiang, J. Y., Yang, J. B., & Zheng, L. (2015). Nondestructive and intuitive determination of circadian chlorophyll rhythms in soybean leaves using multispectral imaging. *Scientific Reports*, 5. <https://doi.org/10.1038/srep11108>
- Patias, P. (n.d.). *Applications, case studies and best practices*.
- Peng, Y., Dallas, M. M., Ascencio-Ibáñez, J. T., Hoyer, J. S., Legg, J., Hanley-Bowdoin, L., Grieve, B., & Yin, H. (2022). Early detection of plant virus infection using multispectral imaging and spatial-spectral machine learning. *Scientific Reports*, 12(1), 3113. <https://doi.org/10.1038/s41598-022-06372-8>
- Peng, Y., Zhang, M., Xu, Z., Yang, T., Su, Y., Zhou, T., Wang, H., Wang, Y., & Lin, Y. (2020). Estimation of leaf nutrition status in degraded vegetation based on field survey and hyperspectral data. *Scientific Reports*, 10(1). <https://doi.org/10.1038/s41598-020-61294-7>
- Reuter, F. (2011). Transformaciones de la imagen: clasificaciones supervisadas y no supervisadas. Facultad de Ciencias Forestales. UNSE. 21 pág
- Ríos-Hernández, R. (2021). *Uso de los Drones o Vehículos Aéreos no Tripulados en la Agricultura de Precisión* *Use of Drones or Unmanned Aerial Vehicles in Precision Agriculture* (Vol. 11, Issue 4, pp. 1–11). <https://orcid.org/0000-0001-6544-1729>



- Saavedra Mora, D., Machado Cuellar, L., Méndez Pastrana, D. A., Murcia Torrejano, V., & Vargas, D. A. (2020). *Manual de interpretación y aplicación de imágenes multiespectrales en cultivos de importancia agrícola en el norte del Huila*.
- Sankaran, S., Khot, L., Espinoza, C., Jarolmasjed, S., Sathuvalli, V., Vandemark, G., Miklas, P., Carter, A., Pumphrey, M., Knowles, N., & Bgood, J. (2015). Low-altitude, high-resolution aerial imaging systems for row and field crop phenotyping: A review. *European Journal of Agronomy*, 70, 112–123. <https://doi.org/10.1016/j.eja.2015.07.004>
- Sarkar, S., Cazenave, A. B., Oakes, J., McCall, D., Thomason, W., Abbott, L., & Balota, M. (2021). Aerial high-throughput phenotyping of peanut leaf area index and lateral growth. *Scientific Reports*, 11(1). <https://doi.org/10.1038/s41598-021-00936-w>
- Sarmiento, A., Suárez Gélvez, J., & Mosquera, J. (2017). *NATURALEZA Y SOCIEDAD: RELACIONES Y TENDENCIAS DESDE UN ENFOQUE EUROCÉNTRICO*. 44, 348–371.
- Stafford, J. v. (2000). Implementing precision agriculture in the 21st century. *Journal of Agricultural and Engineering Research*, 76(3), 267–275. <https://doi.org/10.1006/jaer.2000.0577>
- Stevenson, J. R., Villoria, N., Byerlee, D., Kelley, T., & Maredia, M. (2013). Green Revolution research saved an estimated 18 to 27 million hectares from being brought into agricultural production. *Proceedings of the National Academy of Sciences of the United States of America*, 110(21), 8363–8368. <https://doi.org/10.1073/pnas.1208065110>
- Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. *Computational Intelligence and Neuroscience*, 2016. <https://doi.org/10.1155/2016/3289801>
- Sulecki, P. C. J. (2018). *AGRICULTURA INTELIGENTE*.
- Toribio, G. (15 de Abril de 2021). *Acerca de nosotros: TYC GIS Formación*. Obtenido de TYC GIS: <https://www.cursosteledeteccion.com/ndvi-que-es-y-para-que-sirve/>
- Sun, Y., Liu, Y., Wang, G., & Zhang, H. (2017). Deep Learning for Plant Identification in Natural Environment. *Computational Intelligence and Neuroscience*, 2017. <https://doi.org/10.1155/2017/7361042>
- Verger, A., Vigneau, N., Chéron, C., Gilliot, J.-M., Comar, A., & Baret, F. (2014). Green area index from an unmanned aerial system over wheat and rapeseed crops. *Remote Sensing of Environment*, 152, 654–664. <https://doi.org/https://doi.org/10.1016/j.rse.2014.06.006>
- Viera-Maza, G. (2018). *APLICACIÓN DE PROCESAMIENTO DE IMÁGENES PARA CLASIFICACIÓN DE GRANOS DE CACAO SEGÚN SU COLOR INTERNO*.
- Villanueva-Mejía, D. F. (2018). *ANEXO 1 ANÁLISIS SECTOR AGRÍCOLA Y PECUARIO*.
- Wan, L., Cen, H., Zhu, J., Zhang, J., Zhu, Y., Sun, D., Du, X., Zhai, L., Weng, H., Li, Y., Li, X., Bao, Y., Shou, J., & He, Y. (2020). Grain yield prediction of rice using multi-temporal UAV-based RGB and multispectral images and model transfer – a case study of small farmlands in the South of China. *Agricultural and Forest Meteorology*, 291, 108096. <https://doi.org/https://doi.org/10.1016/j.agrformet.2020.108096>
- Wang, X., Tang, Q., Chen, Z., Luo, Y., Fu, H., & Li, X. (2021). Estimating and evaluating the rice cluster distribution uniformity with UAV-based images. *Scientific Reports*, 11(1). <https://doi.org/10.1038/s41598-021-01044-5>
- Wang, X., Zhang, R., Song, W., Han, L., Liu, X., Sun, X., Luo, M., Chen, K., Zhang, Y., Yang, H., Yang, G., Zhao, Y., & Zhao, J. (2019). Dynamic plant height QTL revealed in maize through remote sensing phenotyping using a high-throughput unmanned aerial vehicle (UAV). *Scientific Reports*, 9(1). <https://doi.org/10.1038/s41598-019-39448-z>
- Wang, Z., Skidmore, A. K., Darvishzadeh, R., & Wang, T. (2018). Mapping forest canopy nitrogen content by inversion of coupled leaf-canopy radiative transfer models from airborne hyperspectral imagery. *Agricultural and Forest Meteorology*, 253–254, 247–260. <https://doi.org/https://doi.org/10.1016/j.agrformet.2018.02.010>
- Xie, C., Shao, Y., Li, X., & He, Y. (2015). Detection of early blight and late blight diseases on tomato leaves using hyperspectral imaging. *Scientific Reports*, 5. <https://doi.org/10.1038/srep16564>

- Yang, B., & Xu, Y. (2021). Applications of deep-learning approaches in horticultural research: a review. In *Horticulture Research* (Vol. 8, Issue 1). Springer Nature. <https://doi.org/10.1038/s41438-021-00560-9>
- Yang, G., Liu, J., Zhao, C., Li, Z., Huang, Y., Yu, H., Xu, B., Yang, X., Zhu, D., Zhang, X., Zhang, R., Feng, H., Zhao, X., Li, Z., Li, H., & Yang, H. (2017). Unmanned Aerial Vehicle Remote Sensing for Field-Based Crop Phenotyping: Current Status and Perspectives. *Frontiers in Plant Science*, 8. <https://doi.org/10.3389/fpls.2017.01111>
- Yang, W., Feng, H., Zhang, X., Zhang, J., Doonan, J. H., Batchelor, W. D., Xiong, L., & Yan, J. (2020). Crop Phenomics and High-Throughput Phenotyping: Past Decades, Current Challenges, and Future Perspectives. In *Molecular Plant* (Vol. 13, Issue 2, pp. 187–214). Cell Press. <https://doi.org/10.1016/j.molp.2020.01.008>
- Zarco-Tejada, P. J., González-Dugo, V., & Berni, J. A. J. (2012). Fluorescence, temperature, and narrow-band indices acquired from a UAV platform for water stress detection using a micro-hyperspectral imager and a thermal camera. *Remote Sensing of Environment*, 117, 322–337. <https://doi.org/https://doi.org/10.1016/j.rse.2011.10.007>
- Zhang, C., and Kovacs, J.M. (2012) The Applications of Small Unmanned Aerial Systems for Precision Agriculture: A Review. *Precision Agriculture*, 13, 693-712.

## **7. ANEXOS.**

7.1 Artículo

7.2 Interfaz modificada en MATLAB.

7.3 Código fuente y recursos de algoritmo de la Red Neuronal Convolutacional (CNN).

7.4 Código fuente y recursos de algoritmo de la Maquina de Soporte Vectorial (SVM).