

METODOLOGÍA PARA LA DETECCIÓN DE SIGATOKA NEGRA USANDO IMÁGENES DIGITALES

Ingeniero Cristian Andrés Escudero Zapata

Proyecto de grado presentado como requisito parcial
Para aspirar al título de Magíster en Ingeniería Eléctrica

Director

Ing. M. Sc Andrés Felipe Calvo

UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE INGENIERÍAS EEFC
PROGRAMA DE MAESTRÍA EN INGENIERÍA ELÉCTRICA
LÍNEA DE AUTOMÁTICA

PEREIRA

2019

El presente trabajo de grado hace parte del proyecto “Sistema experto para la detección automática de la sigatoka negra producida por el hongo *Mycosphaerella fijiensis* en los cultivos comerciales de plátano para la generación de alertas tempranas utilizando TICs”, financiado por el ministerio de educación nacional contrato 160-2018 código 511-3-024-15.

Dedicado a mis padres Humberto Escudero y Vicenta Zapata por el apoyo, amor incondicional y los grandes valores que me han inculcado, todo lo que soy es gracias a ellos.

A Maria, Lilia y Adela por ser mi gran inspiración.

Agradecimientos

Agradezco a todos los integrantes de la familia GIE, en especial a Calvo, Arley, Leidy, Castillo, Santiago por mostrarme este hermoso camino y ayudarme en mi formación académica, profesional y personal. A mis amigos por el apoyo incondicional.

A la Universidad Tecnológica de Pereira por estos 7 años de aprendizaje, soy un profundo convencido de la educación como principal camino para mejorar como sociedad.

Finalmente quiero agradecer a Don German Pulgarin, por abrirnos las puertas de su finca y permitir el desarrollo de este trabajo.

Contenido

1. PREELIMINARES	11
1.1. Introducción	11
1.2. Definición del problema.....	11
1.3. JUSTIFICACIÓN	15
1.4. OBJETIVOS.....	16
1.4.1. Objetivo general.....	16
1.4.2. Objetivo específico.....	16
2. MARCO REFERENCIAL	16
2.1. ESTADO DEL ARTE	16
2.2. MARCO CONCEPTUAL.....	20
2.2.1 Base de datos	20
2.2.2 Procesamiento de imágenes.....	20
2.2.3 Extracción de características.....	20
2.2.4 Técnicas de reducción de dimensionalidad	20
2.2.5 Clasificación.....	21
2.2.6 Redes convolucionales	21
2.3. MARCO TEÓRICO	21
2.3.1 Transformada Rápida de Fourier	21
2.3.2 Transformada Wavelet Discreta (DWT)	22
2.3.4 Histograma	22
2.3.5 Operación de Convolución	23
2.3.6 Espacios de Color	23
2.3.7 Espacio de color TSL.....	23
2.3.8 Espacio de color HSV	24
2.3.9 Espacio de color LAB	25
2.3.10 Escala de Fouré.....	25

2.3.11	Máquinas de vectores de soporte (SVM)	26
2.3.12	Correlación	27
2.3.13	Arboles de Decisión	28
2.3.14	Métodos de clasificación Ensamblados	28
2.3.15	KNN	29
3.	DESARROLLO	29
3.1	METODOLOGÍA DE CLASIFICACIÓN USANDO APRENDIZAJE DE MÁQUINA CLÁSICO	29
3.1.1	Base de datos	30
3.1.2	Procesamiento de imagen	31
3.1.3	Algoritmo de acondicionamiento de imagen	31
3.1.4	Conversión a espacios de color	32
3.1.5	Extracción de descriptores	33
3.1.6	Cálculo del descriptor	33
3.1.7	Construcción de la matriz de características	34
3.1.8	Análisis de dimensionalidad	35
3.1.9	Clasificación de la enfermedad	37
3.2	METODOLOGÍA DE CLASIFICACIÓN USANDO APRENDIZAJE PROFUNDO	38
3.2.1	Acondicionamiento de imagen para clasificación	39
3.2.2	Modelo Red Neuronal Convolutiva	39
3.2.3	Generación, conteo y clasificación de etiquetas	41
3.2.4	Entrenamiento de Modelos de clasificación	42
3.2.5	Entrenamiento del Árbol de decisión	43
3.2.6	Base de datos para entrenamiento red neuronal	43
	Métodos de prueba	44
4.	ANÁLISIS Y RESULTADOS	46

5. CONCLUSIONES.....	57
5.1 Metodología aprendizaje de máquina clásico.....	58
5.2 Metodología usando aprendizaje profundo.....	59
6. TRABAJOS FUTUROS	60
7. REFERENCIAS	60

Índice de Tablas

Tabla 1. Parámetros de entrenamiento Red Neuronal Convolutacional.	42
Tabla 2. Eficiencia de clasificación vs número de bins.	47
Tabla 3. Eficiencia vs cantidad de vecinos.	48
Tabla 4. Comparación de tiempos de las pruebas realizadas.	50
Tabla 5. Matriz de confusión clasificador Ensemble Boosted Tree simple, entrenado con X completo.	51
Tabla 6. Matriz de confusión clasificador Ensemble Boosted Tree jerárquico, entrenado con X completa.	51
Tabla 7. Comparación diagonal principal matrices de confusión de los modelos de clasificación usando el conjunto de datos completo.	52
Tabla 8. Matriz de confusión clasificador SVM, entrenado con X completa.	52
Tabla 9. Matriz de confusión clasificador Árbol de decisión, entrenado con X completa.	53
Tabla 10. Matriz de confusión clasificador KNN, entrenado con X completa.	53
Tabla 11. Comparación diagonal principal matrices de confusión de los modelos de clasificación usando el conjunto de datos reducido.	54
Tabla 12. Matriz de confusión clasificador Ensemble Boosted Tree, entrenado con X reducida.	54
Tabla 13. Matriz de confusión clasificador SVM, entrenado con X reducida.	54
Tabla 14. Matriz de confusión clasificador Árbol de decisión, entrenado con X reducida.	55
Tabla 15. Matriz de confusión clasificador KNN, entrenado con X reducida.	55
Tabla 16. Matriz de confusión Red Neuronal Convolutacional Lenet con 3 clases.	55
Tabla 17. Matriz de confusión Árbol de decisión, resultado de la metodología con aprendizaje profundo.	56
Tabla 18. Matriz de confusión Red Neuronal Convolutacional Lenet con 4 clases.	56

Tabla 19. Matriz de confusión Árbol de decisión, resultado de la metodología con aprendizaje profundo con 4 clases.....	57
---	----

Índice de Figuras

Figura 1. Escala de Fouré.....	26
Figura 2. Diagrama metodología propuesta con aprendizaje de máquina clásico...	30
Figura 3. Construcción de la base de datos.....	31
Figura 4 Etapa procesamiento de imagen.....	31
Figura 5. Algoritmo de acondicionamiento de imagen.....	32
Figura 6. Proceso extracción de descriptores.	33
Figura 7. Procedimiento análisis de dimensionalidad.....	35
Figura 8. Esquema clasificador jerárquico.....	38
Figura 9. Metodología de aprendizaje profundo.	38
Figura 10. Diagrama bloques acondicionamiento de imagen para clasificación usando aprendizaje profundo.	39
Figura 11. Topología Lenet.	41
Figura 12. Algoritmo Conteo de etiquetas.	42
Figura 13. Base de datos para el Entrenamiento de la CNN.	43
Figura 14. Prueba espacios de color.....	46
Figura 15. Correlación matriz de características.	47
Figura 16. Eficiencia vs número de características para K vecinos	48
Figura 17. Análisis de dimensionalidad a nivel de características (a) y a nivel de descriptores (b).....	49

1. PREELIMINARES

1.1. Introducción

1.2. Definición del problema

En Colombia el plátano es el segundo cultivo con mayor cantidad de hectáreas sembradas y el departamento de Risaralda no es la excepción [1]. A lo largo de los años se ha mantenido como sustento tradicional de la economía campesina, que es comprendida principalmente por pequeños agricultores [2]. El plátano además es uno de los principales componentes de la dieta alimenticia en Colombia [3]. En los últimos años la cantidad de área sembrada aumentó significativamente, sin embargo, no se generaron mejoras en el rendimiento o la calidad del producto, esto se genera principalmente por dos factores [4].

El primero es la gran brecha entre los campesinos y las herramientas tecnológicas [5], a pesar del crecimiento exponencial de la tecnología en la última década, no se ha logrado adaptar completamente al sector agrícola. Esto se debe a los altos costos que poseen los sistemas de adquisición y procesamiento de datos idóneos para soportar las condiciones de un cultivo y que un pequeño agricultor de escasos recursos no puede pagar [6] además de los costos de capacitación y mantenimiento que requieren este tipo de sistemas.

Actualmente, los procesos de globalización han incrementado el uso de computadores y smartphone por parte de los campesinos. Por esta razón se han desarrollado aplicativos que guían el proceso de muestreo de Sigatoka a partir de métodos estandarizados como el muestreo de Stover [7]. Sin embargo, también es necesario conocer el estadio de la enfermedad y generar un diagnóstico a partir del mismo.

Desafortunadamente dadas las limitaciones tecnológicas los agricultores se limitan a generar criterios subjetivos que poseen un alto error[8].

El segundo factor que dificulta la producción agrícola es la alta variedad de enfermedades que afectan el cultivo y que se propagan fácilmente debido a las condiciones climáticas propias de una región tropical como Colombia. La enfermedad que más afecta el plátano es la Sigatoka Negra [9], que puede causar pérdidas de hasta el 100% de la producción de no realizarse un control adecuado. Se han desarrollado estudios patológicos que caracterizan la enfermedad de acuerdo a cambios de color y textura que produce en la hoja. A partir de estos se generan métricas que determinan el estadio de la enfermedad en una planta [10]. Otros estudios asocian el crecimiento de la enfermedad a factores climáticos [11], es por esto que algunas metodologías abordan el problema de la detección de la enfermedad usando sensores que miden variables agroclimáticas, donde se aplican modelos de regresión lineal [12], arboles de regresión aditiva y regresión por máquinas de soporte vectorial [13] [14]. Sin embargo, estas mediciones solo permiten determinar la probabilidad de ocurrencia de un evento a futuro.

Otra perspectiva que aborda la detección de la enfermedad es el aprovechamiento de los cambios de color y textura a través de técnicas procesamiento de imágenes y de visión por computador. Sin embargo, utilizar imágenes digitales conlleva a problemas de cambios de iluminación y ruido que se generan al capturar muestras en condiciones no controladas. Por esto en la literatura se plantea tomar las muestras en invernaderos [12] o través de escáneres de laboratorio [15]. Extraer el objeto de estudio de su ambiente natural genera modelos teóricos con altas eficiencias sin embargo, se omite todo el ruido producto de tomar las muestras en campo. Por otro lado, los métodos que si plantean la toma en campo solo generan modelos binarios que determinan si una hoja se encuentra sana o enferma,

omitiendo el estadio de la enfermedad y en la mayoría de casos no permite una detección temprana que evite generar pérdidas en el cultivo. Algunos enfoques utilizan únicamente técnicas de procesamiento de imágenes para realizar el cálculo del promedio ponderado de infección o el porcentaje de área afectada, esto se logra segmentando la enfermedad del resto de la imagen y contando la cantidad de píxeles enfermos para determinar un porcentaje de área afectada. Aunque estos métodos logran resultados interesantes, no se contempla un proceso de calibración de cámara, necesario para obtener distancias y tamaños [16].

Se han desarrollado metodologías más robustas que utilizan visión por computador y que típicamente plantean etapas de segmentación, extracción de características y clasificación a partir de técnicas de aprendizaje de máquina [17]. El principal objetivo en las etapas de segmentación es resaltar rasgos de la enfermedad, buscando aumentar la separabilidad entre los datos. Para esto se plantea el uso de filtros que suavizan la imagen [18], facilitando la segmentación a partir de umbrales [19] o análisis de distribución de densidad [20].

Después de procesada la imagen se realizan etapas de extracción de características de color a partir de momentos estadísticos de sistemas de representación como TSL, HSV LAB, también es común el uso de características de textura como contraste, correlación, homogeneidad y transformadas como la FFT o DWT [21] sin embargo es necesario probar estos descriptores para los diferentes espacios representación mencionados.

Debido a la alta dimensionalidad que generan algunos descriptores, se utilizan técnicas de análisis de características como PCA o Relief que permiten disminuir el tamaño de las matrices de características sin afectar la eficiencia de los clasificadores [15] [22].

Las metodologías para detección de Sigatoka que plantean el uso de técnicas de aprendizaje de máquina implementan modelos de clasificación como redes neuronales, K vecinos, máquinas de soporte vectorial, arboles de decisión [23] [24] [25]. Estos clasificadores reportan altos porcentajes de eficiencia, sin embargo, solo se clasifican imágenes sanas o enfermas. Otros estudios que si plantean determinar el estadio solo evalúan la enfermedad en las etapas más avanzadas, debido a la dificultad de encontrar un modelo de separabilidad en las primeras etapas de infección. Aunque clasificar la enfermedad solo en estado avanzado no le permite al agricultor generar un control efectivo de la enfermedad.

También existen metodologías para la detección de Sigatoka Negra que plantean el uso de técnicas de aprendizaje profundo como redes neuronales convolucionales, estos modelos no requieren un análisis de características ya que las capas de convolución cumplen con esta función, demostrando ser en muchos casos superiores a las técnicas de aprendizaje de máquina tradicionales, sin embargo, los modelos planteados se limitan a clasificar imágenes sanas de enfermas [26].

Por lo mencionado anteriormente se requiere desarrollar una metodología para la detección automática de Sigatoka Negra en cultivos comerciales de plátano, donde se prueben modelos de clasificación a partir de muestras tomadas en condiciones reales y que permitan determinar el nivel de severidad de una hoja a partir de una escala estandarizada. Además, es necesario desarrollar sistemas que requieran un costo computacional bajo y que permitan a futuro una implementación en un dispositivo móvil de acceso para un agricultor.

1.3.JUSTIFICACIÓN

La agricultura es uno de los sectores más influyentes en la economía del país, sin embargo, esta actividad se lleva a cabo por agricultores con pequeñas parcelas que manejan procesos altamente empíricos, pues no poseen la infraestructura tecnológica necesaria para generar altos niveles de producción bajo un estándar de alta calidad. Esto conlleva a una desventaja frente a las grandes empresas agrícolas internacionales, que traen a Colombia alimento a bajo costo ayudadas por los tratados de libre comercio, debido a esto, es necesario desarrollar sistemas accesibles para los pequeños agricultores que permitan generar competitividad y que estandaricen metodologías que ayudadas por buenas prácticas agrícolas potencien al pequeño campesino y generen competitividad [27] [20].

Recientes investigaciones han permitido que la tecnología impacte de manera positiva en los procesos agronómicos. A partir de variables climáticas es posible determinar la probabilidad de aparición de un evento, relacionando medidas de variables como la temperatura y la humedad relativa con la aparición de Sigatoka negra en cultivos de plátano [8]. Desarrollos en agricultura de precisión usando procesamiento digital de imágenes han logrado segmentar diferentes tipos de enfermedades. Se han propuesto técnicas de segmentación utilizando características de color y textura que podrían aumentar la separabilidad entre clases [19]. Algunos descriptores como las transformadas de Fourier y Wavelet han logrado extraer características relevantes cuando el objeto de estudio presenta cambios de textura [23]. Y a través de estos se entrenan técnicas de aprendizaje de máquina que logran resultados satisfactorios y que podrían ser aplicados en la detección de Sigatoka Negra en cultivos de plátano. Una metodología para la detección de Sigatoka negra en cultivos de plátano utilizando sistemas de adquisición de datos de bajo costo permitiría a los pequeños agricultores acceder a herramientas

tecnológicas que eliminen al máximo el criterio subjetivo producido por el análisis visual realizado por el campesino, mejorando la calidad en la producción de sus cultivos [20] [8] lo que afectaría positivamente el sector agrícola.

1.4. OBJETIVOS

1.4.1. Objetivo general

Desarrollar una metodología para la detección de sigatoka negra en cultivos de plátano usando imágenes digitales.

1.4.2. Objetivo específico

1. Crear una base de datos anotada con imágenes digitales etiquetadas con los niveles de severidad de la enfermedad a partir de una escala estandarizada.
2. Determinar un método de aprendizaje de máquina que permita clasificar la enfermedad.
3. Realizar una validación estadística de la metodología propuesta.

2. MARCO REFERENCIAL

2.1. ESTADO DEL ARTE

El procesamiento digital de señales ha permitido desarrollar sistemas que abordan problemas en diferentes áreas del conocimiento, la agricultura es uno de los campos más susceptible a nuevos avances tecnológicos ya que los procesos llevados a cabo en su mayoría aún son empíricos, a pesar de esto se han generado metodologías que permiten identificar el estado de una planta de acuerdo a una enfermedad en

particular. A continuación, se presentarán los desarrollos más significativos en la detección y clasificación de enfermedades en cultivos.

Se han desarrollado aplicativos donde se explica a detalle el proceso de muestreo en cultivos de plátano utilizando el método de Stover, donde se plantean sistemas de adquisición y análisis de información para el diagnóstico de sigatoka. Sin embargo, la detección de la enfermedad se realiza a criterio del agricultor, lo que conlleva a una medida subjetiva que podría ser errónea debido a una mala percepción [7]. Otros aplicativos permiten el cálculo de un coeficiente para representar la velocidad a la que se desarrolla la enfermedad [28].

Enfoques más avanzados para la detección de enfermedades en plantas no solo presentan una guía, sino también utilizan información muestreada por diferentes sensores agroclimáticos, permitiendo desarrollar sistemas de predicción de enfermedades en cultivos [15]. A priori, las técnicas de vigilancia y pronóstico de Sigatoka, correlacionan el promedio ponderado de infección, la posición de la hoja más joven manchada y la emisión foliar, con mediciones de temperatura, nubosidad, insolación, y radiación utilizando regresión lineal múltiple por mínimos cuadrados [29], aunque generar un modelo lineal no es totalmente conveniente ya que las variables que afectan los cultivos generalmente presentan comportamientos no lineales. Debido a esto, sistemas más robustos utilizan métodos de regresión con modelos como redes neuronales o máquinas de soporte vectorial usando características obtenidas de las mediciones anteriormente mencionadas.

Otros avances, han generado redes de sensores inalámbricos usando módulos Xbee que permiten medir las variables descritas [25]. Los cuales a través de técnicas como lógica difusa en conjunto con fusión de datos permiten además de pronosticar enfermedades en cultivos, calcular probabilidad de eventos como incendios forestales o sequía agrícola en región es áridas, estos aplicativos están planteados

para sistemas operativos Android [30] [31] [32]. Las metodologías que utilizan sensores que miden variables agroclimáticas poseen ventajas sobre las técnicas que extraen información a partir de mediciones de potasio, nitrógeno u otros elementos químicos [33] ya que al ser técnicas no invasivas, el objeto de estudio no es alterado y no necesitan un análisis de laboratorio. Sin embargo si se parte desde un análisis epistemológico fundamentado hacia el desarrollo sostenible se hace necesario el uso de sensores que permitan obtener más información como las cámaras digitales, ligado al procesamiento de imágenes digitales o visión por computador [34].

En este orden de ideas, se han generado métodos para establecer el promedio ponderado de infección utilizando análisis del espacio de color RGB por medio de histogramas [16]. Otras aplicaciones utilizan momentos estadísticos para calcular el ancho y largo de la hoja de plátano, a partir de modelos de regresión lineal por mínimos cuadrados que determinan el área foliar afectada por Sigatoka Negra, sin embargo, las muestras se toman bajo condiciones controladas en un invernadero [18]. Por tanto, estas técnicas solo requieren el uso de un espacio de color como base para la extracción de características pues no prevén los cambios de iluminación que se generan al capturar una imagen en una escena real. A pesar de esto el uso de segmentación por color es ampliamente utilizado, algunos enfoques plantean convertir las imágenes capturadas a espacios de color como $L^*a^*b^*$, HSV, TSL y LAB [20].

Los sistemas más robustos no solo realizan segmentación, sino que implementan técnicas de aprendizaje de máquina utilizando modelos provenientes de máquinas de soporte vectorial y ANFIs que se entrenan a partir de características de momentos estadísticos de color, textura y forma, para determinar si una hoja se encuentra o no infectada por sigatoka negra [19].

Utilizando características similares se plantea la detección mediante árboles genéticos [13]. Otro enfoque común para la clasificación de enfermedades en plantas es la extracción de características del canal HSV unidos a información de texturas y magnitud del gradiente para entrenar modelos como redes neuronales artificiales, máquinas de soporte vectorial, K vecinos más cercanos y árboles extremadamente aleatorios [24] [35] [36]. También se ha planteado la extracción de características a partir de métodos como la transformada Wavelet discreta que son de gran utilidad, ya que ésta detecta los altos cambios de frecuencia generados por las manchas características de la Sigatoka en las hojas. Las características como HOG también han demostrado ser eficientes para el entrenamiento de modelos como Árboles de decisión [25].

Los modelos descritos anteriormente poseen eficiencias mayores a 90% en la clasificación de hojas sanas y enfermas, sin embargo, no tienen en cuenta que la enfermedad se puede encontrar en diferentes estadios en los que se presentan diferentes síntomas visuales y donde el manejo por parte del agricultor debe ser diferente [23].

Existen además metodologías que plantean el aprovechamiento de técnicas de aprendizaje profundo con redes neuronales convolucionales a partir de topologías como LeNet. El uso de estas técnicas simplifica algunos procesos, ya que no se necesita el uso de descriptores, pues las capas de convolución extraen patrones relevantes de la enfermedad, sin embargo, la salida de la red planteada sigue siendo binaria, generando etiquetas de sano y enfermo, que omiten todos los posibles estadios de la enfermedad [26].

2.2. MARCO CONCEPTUAL

2.2.1 Base de datos

Una base de datos es un conjunto de datos ordenados y etiquetados de acuerdo a un contexto, de tal forma que pueda ser accedido por un programa. En el área del aprendizaje de máquina se utilizan para entrenar modelos de clasificación [37].

2.2.2 Procesamiento de imágenes

Conjunto de técnicas que se aplican a imágenes digitales con el objetivo de mejorar la información encontrada en estas, entre las principales técnicas se encuentran el suavizado, filtrado, realce de bordes y segmentación de objetos de interés [38].

2.2.3 Extracción de características

Consiste en la búsqueda de atributos o cualidades que facilitan la identificación de una entidad en particular como fenómenos físicos, comportamientos, movimientos, entre otros. En el caso específico de las ciencias de computación e ingeniería, la extracción de características, se realiza aplicando descriptores matemáticos. Los descriptores matemáticos dependen específicamente de la naturaleza de los datos a analizar [21].

2.2.4 Técnicas de reducción de dimensionalidad

Consiste en un cúmulo de técnicas que buscan reducir la dimensionalidad de un conjunto de variables aleatorias a partir de una representación donde la información se conserve. Entre las más comunes se encuentran técnicas como PCA o Relieff, sin embargo, la utilidad de estos métodos está sujeto a la naturaleza de la base de datos [39].

2.2.5 Clasificación

Es un proceso de estimación que tiene como objetivo encontrar un modelo matemático que permita separar o identificar una entidad en particular utilizando un conjunto de ejemplos o información suministrada. El proceso de clasificación se puede realizar utilizando clasificación supervisada, clasificación no supervisada, aprendizaje por esfuerzo, transducción o aprendizaje multi-tarea. Algunos de los clasificadores más comunes en este tipo de tareas son: los vecinos más cercanos (KNN), las máquinas de vectores de soporte (SVM), los árboles de decisión, las redes neuronales artificiales (RNA), clasificador Bayesiano, árboles de decisión, entre otros [40].

2.2.6 Redes convolucionales

Las redes convolucionales son redes para el procesamiento de datos como series de tiempo e imágenes digitales, su nombre viene debido a que aplican operaciones de convolución que funcionan como extractores de características [41].

2.3. MARCO TEÓRICO

2.3.1 Transformada Rápida de Fourier

Algoritmo que realiza el cálculo de la transformada discreta de Fourier, que consiste en la transformación de señales entre el dominio del tiempo y el dominio de la frecuencia. Es de gran uso en el procesamiento de imágenes y visión por computador para determinar características en una imagen, se define En tiempo discreto [42]:

$$Y_{p+1,q+1} = \sum_{j=0}^{m-1} \sum_{k=0}^{n-1} W_m^{jp} W_n^{kp} X_{j+1,k+1} \quad (1)$$

Donde $W_n = e^{(-2\pi i)/n}$ y $W_m = e^{(-2\pi i)/m}$

2.3.2 Transformada Wavelet Discreta (DWT)

Esta técnica nace como una alternativa para superar los problemas de resolución de la transformada de Fourier de tiempo corto (STFT), haciendo posible una buena representación de una señal tanto en tiempo como en frecuencia de forma simultánea. Básicamente, la idea de la transformada Wavelet es filtrar una señal en el dominio del tiempo mediante filtros paso bajo y paso alto para eliminar ciertas componentes de alta o baja frecuencia de la señal, el procedimiento se repite para las señales resultantes del proceso de filtrado y a esta operación se le denomina descomposición [40].

Dada la función discreta $f_{[n]}$, la cual se define como la señal a analizar, su transformada Wavelet $c_{[i,k]} = f_{[n]} \varphi_{i,j[n]}$, se define como:

$$c_{[i,k]} = f_{[n]} \varphi_{i,j[n]} \quad (2)$$

2.3.4 Histograma

Representación de una variable en forma de bins, que genera una forma rápida para conocer la distribución en frecuencia de los valores representados. Los histogramas son comúnmente usados en visión por computador, ya que permiten determinar características de un objeto de interés [40].

2.3.5 Operación de Convolución

En aplicaciones de aprendizaje de máquina la entrada usualmente es un arreglo multidimensional de datos, la operación de convolución es usada como un kernel que se puede expresar como una sumatoria sobre un número finito de elementos, si la entrada es una imagen se puede expresar como: [41]

$$S_{(i,j)} = (I * K)(i,j) = \sum_m \sum_n I_{(m,n)} K_{(i-m,j-n)} \quad (3)$$

2.3.6 Espacios de Color

Un espacio de color es un modelo matemático en la que los colores pueden representarse como combinaciones, que comúnmente se presentan a partir de tres o cuatro componentes, Entre los más comunes se encuentran los espacios RGB, TSL, LAB,HSV, HSI. Cada espacio de color genera una representación que puede ser usada como descriptor y que a menudo simplifica la identificación y extracción de objetos de una escena [38].

2.3.7 Espacio de color TSL

Espacio de color que se define como tinte, saturación y luminosidad, este se desarrolló principalmente para la detección de rostros para la conversión de RGB a TSL se requieren las siguientes ecuaciones [43]:

$$T = \begin{cases} \frac{\arctan(\frac{r'}{g'})}{2\pi} + \frac{1}{4}, & g' < 0 \\ \frac{\arctan(\frac{r'}{g'})}{2\pi} + \frac{1}{4}, & g' < 0 \\ 0, & g' = 0 \end{cases} \quad (4)$$

$$S = \left[\frac{9}{5} (r'^2 + g'^2) \right]^{\frac{1}{2}} \quad (5)$$

$$L = 0.299R + 0.587G + 0.114B \quad (6)$$

Donde $r' = \left(r - \frac{1}{3} \right)$, $g' = \left(g - \frac{1}{3} \right)$, $r = \frac{R}{R+G+B}$ y $g = \frac{G}{R+G+B}$

2.3.8 Espacio de color HSV

Espacio de color que se define como matiz, saturación y brillo, donde la matriz se representa como un grado de ángulo cuyos valores posibles van de 0 a 360 grados. La saturación representa la distancia al eje de brillo negro-blanco, cuanto menor sea la saturación de un color mayor tonalidad grisácea habrá y más decolorado estará. Y el brillo representa la altura en un eje blanco-negro [44].

$$H = \begin{cases} 0^\circ, & \text{if } C = 0 \\ 60^\circ * \left(\frac{G - B}{C} \right) \text{ mod } 6, & \text{if } M = R \\ 60^\circ * \left(\frac{B - R}{C} \right) + 2, & \text{if } M = G \\ 60^\circ * \left(\frac{R - G}{C} \right) + 4, & \text{if } M = B \end{cases} \quad (7)$$

$$S = \begin{cases} 0, & \text{if } V = 0 \\ \frac{C}{V}, & \text{otherwise} \end{cases} \quad (8)$$

$$V = M \quad (9)$$

Donde

$$M = \max(R, G, B), m = \min(R, G, B), C = M - m$$

2.3.9 Espacio de color LAB

LAB es un espacio de color tridimensional representado por luminosidad de negro a blanco, rojo a verde y gradiente del azul, se usa para describir todos los colores que pueden percibir el ojo humano. Para la conversión de RGB a LAB se plantean las siguientes ecuaciones.

$$L = 0.2126R + 0.7152G + 0.0722 \quad (10)$$

$$A = 1.4749(0.2213R - 0.3390G + 0.1177B) + 128 \quad (11)$$

$$B = 0.6245(0.1949R + 0.6057G - 0.8006B) + 128 \quad (12)$$

2.3.10 Escala de Fouré

La escala de Fouré, es una división estandarizada de la enfermedad Sigatoka Negra, que determina seis estadios a partir de los síntomas visuales de la hoja de plátano [9]. A continuación, se presentan los estados y se ilustra la escala en la figura 1.

- Estado 1: Pequeñas lesiones o puntos de color blanco-amarillento a marrón, de 1mm de longitud, denominadas pizcas, apenas visibles en el envés de la hoja.
- Estado 2: Rayas o estrías cloróticas de 3-4mm de longitud por 1mm de ancho, de color marrón.
- Estado 3: Las rayas o estrías se alargan y amplían dando la impresión de haber sido pintadas con pincel, sin bordes definidos y de color café, que pueden alcanzar hasta 2cm de longitud.
- Estado 4: Manchas ovaladas de color café en el envés y negro en el haz
- Estado 5: Manchas negras rodeadas de un anillo negro y acces un halo amarillento o centro seco semihundido.
- Estado 6: Manchas con centro seco y hundido, de coloración marrón clara, rodeadas de tejido clorótico

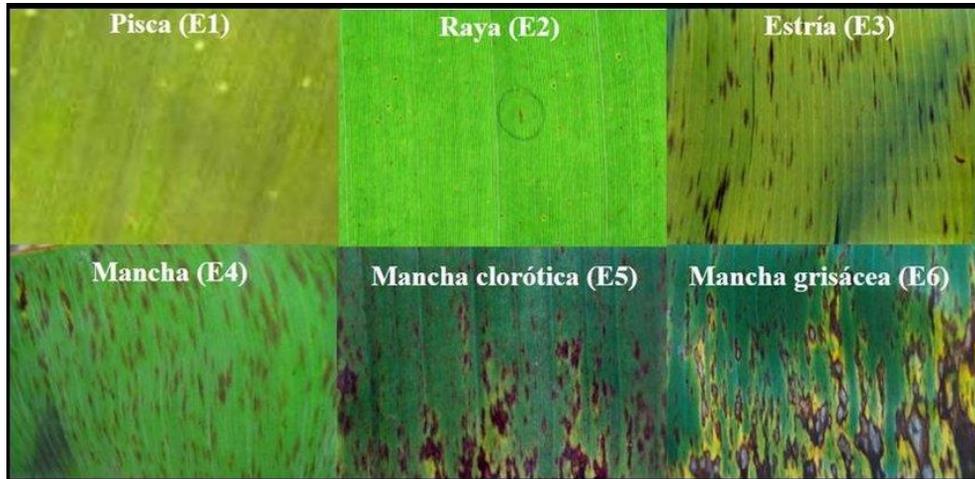


Figura 1. Escala de Fouré.

2.3.11 Máquinas de vectores de soporte (SVM)

Las máquinas de soporte vectorial son una metodología de clasificación asistida basada en la teoría de Vapnik. Este método tiene como objetivo la estimación de un modelo de aprendizaje que permita la clasificación de un evento binario, su formulación matemática depende de la naturaleza de los datos; es decir que existen diferentes formulaciones dependiendo de la linealidad o de la distribución de sus datos. En su forma más sencilla una SVM tiene como objetivo encontrar el hiperplano que separe y maximice el margen entre las clases en este espacio [45].

La SVM se considera un clasificador binario, el cual calcula una etiqueta $y \in \{-1,1\}$ utilizando el siguiente modelo:

$$y = w^T \phi(x) + b \quad (13)$$

Donde, $\phi(x) = R \rightarrow H$ es una transformación del conjunto de entrada a un conjunto con igual o mayor dimensión, lo que se supone que facilita la separación de las clases. El vector w define el hiperplano de separación en espacio $\phi(x)$. b representa el bias o sesgo con respecto al origen de coordenadas. La SVM aborda el problema de clasificación relajando el concepto de margen, para lo que se introduce variables de sesgo $\varepsilon_i \geq 0$, donde la función de costo se define como:

$$c_{(y,f(x,w))} = \begin{cases} 0 & \text{si } y = \text{sing}(w^T \phi(x) + b) \\ 1 & \text{si } y \neq \text{sing}(w^T \phi(x) + b) \end{cases} \quad (14)$$

Donde existe el riesgo empírico $R_{emp} = \sum_{i=1}^n \varepsilon_i^\sigma$, Siendo generalmente $\sigma = 1$ en problema de clasificación. Por lo tanto, el problema de estimación del hiperplano de separación, se define como la solución al siguiente problema de optimización cuadrático (QP):

$$\min_{w,b,\varepsilon} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \varepsilon_i \quad (15)$$

$$\text{Sujeto a } y_i(w^T \phi(x_i) + b) \geq 1 - \varepsilon_i \quad \forall_i = 1 \dots, n \quad (16)$$

Donde, x_i es una muestra del conjunto de entrenamiento con su respectiva etiqueta y C se define como una constante de regularización. Buscando el hiperplano óptimo en (6), se requiere resolver un problema QP, que puede ser resuelto construyendo un Lagrangiano y transformándolo en el dual. Por lo que el problema de optimización se define como:

$$\max_{\alpha} LD = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \phi^T(x_i) \phi(x_j) \quad (17)$$

$$\text{Sujeto a } \sum_{j=1}^n \alpha_j y_j = 0 \quad (18)$$

$$0 \leq \alpha_i \leq C \quad \forall_i = 1 \dots, n \quad (19)$$

2.3.12 Correlación

Mediante observación directa se puede determinar de manera cualitativa la relación entre variables aleatorias, si X y Y son las dos variables aleatorias en consideración, si Y tiende a aumentar a medida que X aumenta, se dice que la correlación es positiva o directa, Si Y tiende a disminuir a medida que X aumenta, se dice que es una correlación negativa o inversa, si no parece haber relación entre las variables, se dice que no hay relación entre ellas, es decir, están descorrelacionadas [46].

La correlación más comúnmente usada es la correlación de Pearson, para una columna X_a en una matriz X y una columna Y_b en una matriz Y , teniendo la media

$\bar{X}_a = \sum_{i=1}^n \frac{(X_{a,i})}{n}$ y $\bar{Y}_b = \sum_{j=1}^n \frac{(X_{b,j})}{n}$ la correlación está definida en la ecuación 20 como sigue.

$$rho(a, b) = \frac{\sum_{i=1}^n (X_{a,i} - \bar{X}_a)(Y_{b,i} - \bar{Y}_b)}{\{\sum_{i=1}^n (X_{a,i} - \bar{X}_a)^2 \sum_{j=1}^n (Y_{b,j} - \bar{Y}_b)^2\}^{1/2}} \quad (20)$$

2.3.13 Árboles de Decisión

Los árboles de decisión son algoritmos versátiles de aprendizaje automático que realizan tareas de clasificación y regresión, estos permiten ajustar conjuntos de datos complejos, también son componentes fundamentales de los árboles aleatorios que se encuentran entre los algoritmos más potentes disponibles en la actualidad. Dado un conjunto de datos se fabrican diagramas de construcciones lógicas, muy similares a los sistemas de predicción basados en reglas, que sirven para representar y categorizar una serie de condiciones que ocurren de forma sucesiva, para la resolución de un problema. Los árboles están formados por nodos que toman una decisión ante varias opciones posibles [47].

2.3.14 Métodos de clasificación Ensamblados

Los métodos de clasificación ensamblados son un conjunto de algoritmos de aprendizaje de máquina que realizan una ponderación de etiquetas a partir de un conjunto finito de clasificadores simples generando una hipótesis, sin embargo esta no necesariamente está contenida dentro del espacio de etiquetas de los modelos a partir de los cuales se construye, por tanto se puede demostrar que los conjuntos tienen más flexibilidad en las funciones que pueden presentar, esta flexibilidad en la práctica tiende a reducir los problemas relacionados con el sobreajuste de datos de entrenamiento.

Los conjuntos tienden a producir mejores resultados cuando existe una diversidad significativa de modelos [47].

2.3.15 KNN

KNN es un método de clasificación no paramétrico usado para clasificación y regresión, la entrada consiste en K ejemplos de entrenamiento más cercanos en el espacio de características, en clasificación la salida es una membresía de la clase, un objeto se clasifica por una pluralidad de votos de sus vecinos y el objeto se asigna a la clase más común entre sus K vecinos más cercanos, los vecinos se toman de un conjunto de objetos para los que se conoce la clase o propiedad del objeto [48].

3. DESARROLLO

3.1 METODOLOGÍA DE CLASIFICACIÓN USANDO APRENDIZAJE DE MÁQUINA CLÁSICO

En esta sección se describe a detalle el trabajo realizado, inicialmente se presenta la metodología de trabajo propuesta, cuyo objetivo es entrenar un método de aprendizaje de máquina tradicional que clasifique la enfermedad sigatoka negra en 4 niveles de severidad usando imágenes digitales. En la figura 2 se observa el diagrama metodológico que se explicará a detalle. Después se presenta una metodología de clasificación utilizando técnicas de aprendizaje profundo (ver figura 9) con la cual se realizará una comparación.

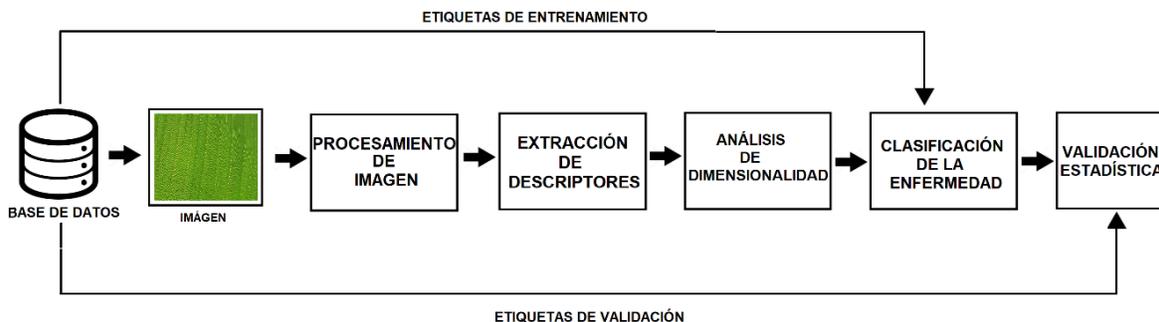


Figura 2. Diagrama metodología propuesta con aprendizaje de máquina clásico.

3.1.1 Base de datos

En la literatura existen bases de datos con imágenes de la enfermedad Sigatoka Negra, infortunadamente, no fueron construidas bajo un protocolo de captura y muestreo adecuado que permitan clasificar las hojas en diferentes niveles de severidad, por otro lado, éstas poseen un número reducido de muestras obviando relevancia estadística necesaria para entrenar modelos de aprendizaje de máquina o los autores no facilitan el acceso a estas bases de datos [8] [29]. Lo que imposibilita aplicar una metodología estructurada como la propuesta en este artículo con el fin de replicar y comparar métodos similares. Debido a lo anterior se construye una base de datos anotada que cumpla con las características requeridas.

Como se observa en la figura 3. Se capturaron imágenes en cultivos de plátano y se etiquetaron en 4 clases según la escala de Fouré. Las muestras se capturaron con celulares cuyas cámaras poseen una resolución de 9.6MP (4128x2322) y 13Mp (3120x4160).

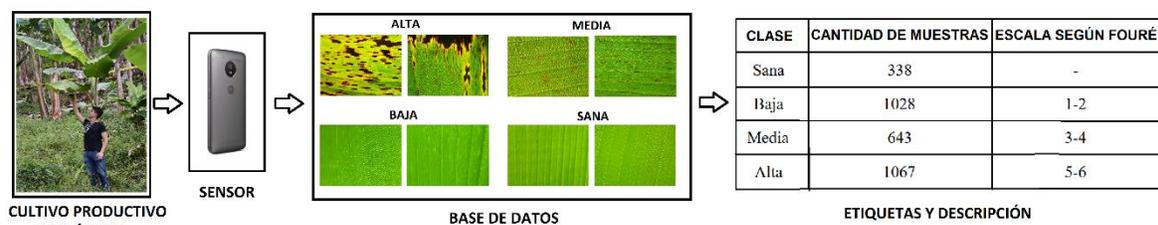


Figura 3. Construcción de la base de datos.

3.1.2 Procesamiento de imagen

Este instrumento metodológico procesa las imágenes con el objetivo de resaltar patrones y características relevantes. El diagrama de bloques que resume el procedimiento se enseña la figura 4, el cual se conforma por etapas de acondicionamiento de imagen y transformaciones a diferentes espacios de color, a continuación, se describe cada uno de estos.

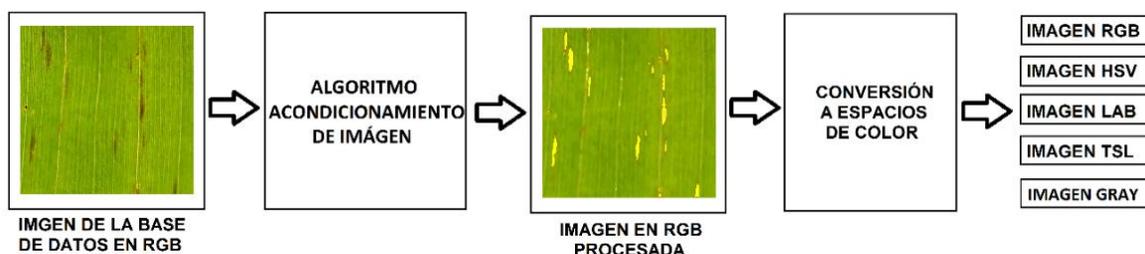


Figura 4 Etapa procesamiento de imagen.

3.1.3 Algoritmo de acondicionamiento de imagen

Se implementa un algoritmo de procesamiento de imágenes que resalta los píxeles donde se encuentra la enfermedad. Como se observa en la figura 5, primero se aplica un filtro gaussiano. Después se realiza una conversión de RGB a HSV y se selecciona el canal V. La selección del espacio de color y del canal se realizó a partir de una validación experimental utilizando una muestra de la base de datos que se evidencia

en la sección de resultados. Después se divide la imagen en ventanas de 500x500 píxeles. Para cada ventana se establece un umbral de $\mu \pm 3\sigma$, donde todo píxel fuera de este rango se considera enfermo y se le aplica una ganancia de 3 saturando su valor. Por último, se unen las ventanas y se regresa al espacio RGB. En la figura 4 se observa el resultado de aplicar el algoritmo propuesto.

```

// Resaltado de pixeles enfermos

Entrada: Imagen en espacio RGB ( $I_{RGB}$ )
Salida: Imagen RGB con pixeles enfermos resaltados ( $I_{Resaltada}$ )

Inicio
    Aplicar filtro espacial Gaussiano a  $I_{RGB}$ 
     $I_{HSV} \leftarrow rgb2hsv(I_{RGB})$ 
     $I_V \leftarrow I_{HSV}(:, :, 3)$ 
    Dividir  $I_V$  en ventanas  $I_v$  de 500x500 píxeles
    Calculo de media ( $\mu$ ) y desviación estándar ( $\sigma$ ) a cada  $I_v$ 
    Para cada  $I_v$ 
        
$$I_{v(i,j)} = \begin{cases} I_{v(i,j)} * 3, & \text{Si } (I_{v(i,j)} > \mu + 3\sigma) \text{ ó } (I_{v(i,j)} < \mu - 3\sigma) \\ I_{v(i,j)}, & \text{En otro caso} \end{cases}$$

    Unir ventanas de  $I_v$  en  $I_V$ 
     $I_{HSV}(:, :, 3) \leftarrow I_V$ 
     $I_{Resaltada} = hsv2rgb(I_{HSV})$ 

Fin

rgb2hsv() = Conversión de espacio color RGB a HSV
hsv2rgb() = Conversión de espacio color HSV a RGB

```

Figura 5. Algoritmo de acondicionamiento de imagen.

3.1.4 Conversión a espacios de color

Una vez se encuentran acondicionadas las muestras, buscando encontrar diferentes espacios de representación de la enfermedad, como se enseña en la figura 4 se convierte la imagen a los espacios de color más usados en la literatura para la detección de enfermedades en plantas que son RGB, TSL, LAB, HSV, GRAY [21].

3.1.5 Extracción de descriptores

En esta etapa se computan los principales descriptores que reporta el estado del arte para identificar patrones en una imagen que tiene variaciones de iluminación producto de un entorno no controlado como lo es un cultivo [49] [35]. Este proceso no solo se extraen las propiedades de color en la imagen sino también transformadas espacio frecuencia generando un proceso robusto a cambios de iluminación.

La extracción de descriptores consta de dos partes fundamentales, en la primera parte se calculan los vectores x formados por los términos [DWT, F, H] para cada canal de los espacios de representación utilizados, en la figura 6 se observa este proceso para cada muestra de la base de datos. En segundo lugar, se concatenan los vectores obtenidos y se construye la matriz de características X .

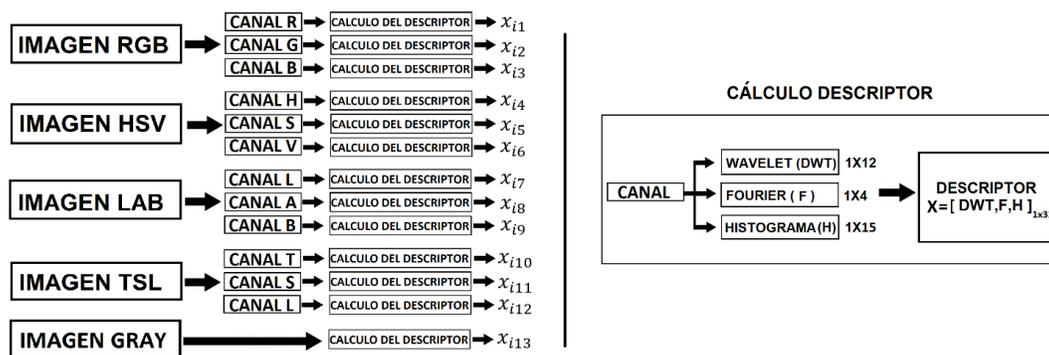


Figura 6. Proceso extracción de descriptores.

3.1.6 Cálculo del descriptor

La metodología propuesta comprende el estudio de tres descriptores que se explican a continuación.

El primer descriptor es la Transformada Discreta Wavelet (DWT) en dos dimensiones usando la familia Symlets de cuarto orden, de esta se extraen las matrices de

coeficientes horizontales (j), verticales (m) y diagonales (u) [50]. A cada matriz se le calculan los momentos estadísticos de media (u), desviación estándar (σ), sesgo (δ) y curtosis (κ) obteniendo el descriptor $DWT_{(j,m,u)}$ dado por:

$$DWT_{(j,m,u)} = [u_j, u_m, u_u, \sigma_j, \sigma_m, \sigma_u, \delta_j, \delta_m, \delta_u, \kappa_j, \kappa_m, \kappa_u] \quad (21)$$

El segundo descriptor es la Transformada Rápida de Fourier en dos dimensiones [51], que computa la matriz (y). A esta matriz se le calculan los mismos momentos estadísticos de DWT, obteniendo así el descriptor F_y dado por:

$$F_{(y)} = [u_j, \sigma_j, \delta_u, \kappa_u] \quad (22)$$

El último cálculo realizado en esta etapa es un histograma con 15 bins [52], de donde se obtiene el descriptor H. La elección del número de bins se presenta en la sección de resultados. Finalmente se obtiene el vector de datos x_i , que resulta de concatenar los tres descriptores propuestos para cada sistema de representación propuesto, así:

$$x_i = [DWT_{(j,m,u)}, F_y, H]_{1 \times 31} \quad (23)$$

Donde i es el canal de color a computar.

3.1.7 Construcción de la matriz de características

Obtenidos los trece vectores se construye la matriz de características X, definida por

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13...} & x_{1m} \\ x_{21} & x_{22} & x_{23...} & x_{2m} \\ x_{31} & x_{32} & x_{33...} & x_{3m} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & x_{n3...} & x_{n,m} \end{bmatrix} \quad (24)$$

Donde n corresponde al número de ejemplos de la base datos y m es la cantidad de vectores computados. Lo anterior permite obtener una matriz de 3071 muestras con

403 características. A continuación, se presenta la concatenación de una fila, que corresponde a una muestra de la matriz X.

$$[x_R, x_G, x_B, x_T, x_S, x_L, x_L, x_A, x_B, x_H, x_S, x_V, x_{GRAY}] \quad (25)$$

3.1.8 Análisis de dimensionalidad

De la etapa anterior se obtiene un conjunto de 3071x403 elementos que suponen un alto costo computacional, dado que se busca desarrollar una metodología con condiciones aptas para ser implementada en dispositivos con baja capacidad de procesamiento, se realiza un análisis de descriptores que permitirá reducir la dimensionalidad del espacio de representación. En la figura 7 se ilustra un diagrama de bloques de esta etapa.



Figura 7. Procedimiento análisis de dimensionalidad.

Inicialmente se calcula la correlación de la matriz X, esto genera un método de visualización que permite magnificar el traslape de la base de datos y así determinar si los descriptores utilizados resaltan el patrón de la enfermedad, para mayor información consulte la documentación de [53]. En la figura 15 se evidencia que algunas características poseen una alta correlación, por tanto, no aportan información distintiva de la enfermedad. Además, se observan características con valor cero o indeterminado. Es necesario eliminar estas columnas de la matriz y

evitar posibles errores en el entrenamiento del modelo. La figura 10 también evidencia la necesidad de aplicar una técnica de reducción de dimensiones con el objetivo de disminuir el traslape de los datos.

Debido a lo anterior, se analizan los pesos calculados por Relief para determinar el aporte de información a nivel de características y de descriptores, después se realiza una comparación respecto [54]. Esto tiene como objetivo encontrar una combinación de descriptores que generen un menor costo computacional y una eficiencia de acierto adecuada.

La primera métrica comprende el estudio tradicional de Relief a nivel de característica, donde se computa una ponderación del peso propuesto para cada característica basada en sus k vecinos más cercanos. Se usa este método debido a que es preciso al ser aplicado en espacios con dependencia entre características y un alto nivel de ruido [55].

La segunda métrica es una técnica heurística a partir de la cual se estudia el aporte de los descriptores DWT, F y H para cada canal, esto se realiza sumando el valor obtenido por Relief para cada una de las características del descriptor, tal como se observa en la ecuación 26.

$$PesoDescriptor = \sum_{i=1}^n PesoCaracteristica_{(i)} \quad (26)$$

Donde n es la cantidad de características que posee cada descriptor y $PesoCaracteristica$ el peso computado por Relief, los descriptores que obtengan mayor magnitud, permitirían obtener información relevante para la clasificación y se incluirán la cantidad necesaria para alcanzar una eficiencia similar a la obtenida con todo el conjunto completo de datos. Esto se realiza debido a que se busca reducir la

cantidad de descriptores a computar, lo que no siempre se logra con los métodos tradicionales de reducción de características.

La última métrica utiliza el criterio planteado en [54]. Donde se afirma que los patrones de textura son indiferentes al color. Lo anterior supondría que no es necesario calcular los descriptores W y F en los espacios de color propuestos, lo que reduciría el costo computacional requerido.

3.1.9 Clasificación de la enfermedad

En esta etapa se busca establecer un método de clasificación adecuado, como modelo de clasificación se propone utilizar la técnica de aprendizaje de máquina Ensemble Boosted Tree usando la metodología Adaptive Boosting, con 30 ciclos de aprendizaje en conjunto. Este método ensamblado permite generar un esquema robusto de árboles de decisión sin incrementar drásticamente el costo computacional [56].

La base de datos usada para el entrenamiento del modelo posee un desbalance intrínseco debido a la naturaleza de la enfermedad. En estos casos el estado del arte sugiere utilizar métodos de clasificación ponderados, por tanto, se utiliza un clasificador jerárquico. Para esto se entrena un modelo biclase que clasifique muestras en estado “Alta” y “No Alta”. Después se entrena otro modelo con enfoque multiclase utilizando una cantidad de muestras balanceadas que genera etiquetas “Media”, “Baja”, “Sana”. Lo anterior evita generar clasificadores sesgados a las clases con mayor número de muestras. En la figura 8 se observa la jerarquía del clasificador.

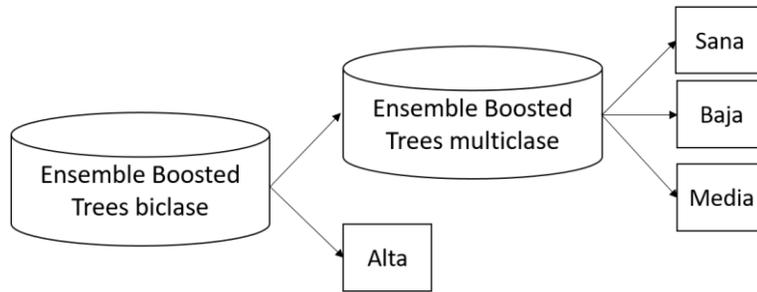


Figura 8. Esquema clasificador jerárquico.

3.2 METODOLOGÍA DE CLASIFICACIÓN USANDO APRENDIZAJE PROFUNDO

Esta metodología se compone de etapas de acondicionamiento de imágenes para ser procesadas por una red neuronal convolucional, entrenamiento de los parámetros de la red, generación y conteo de etiquetas y validación, como se observa en la figura 9.

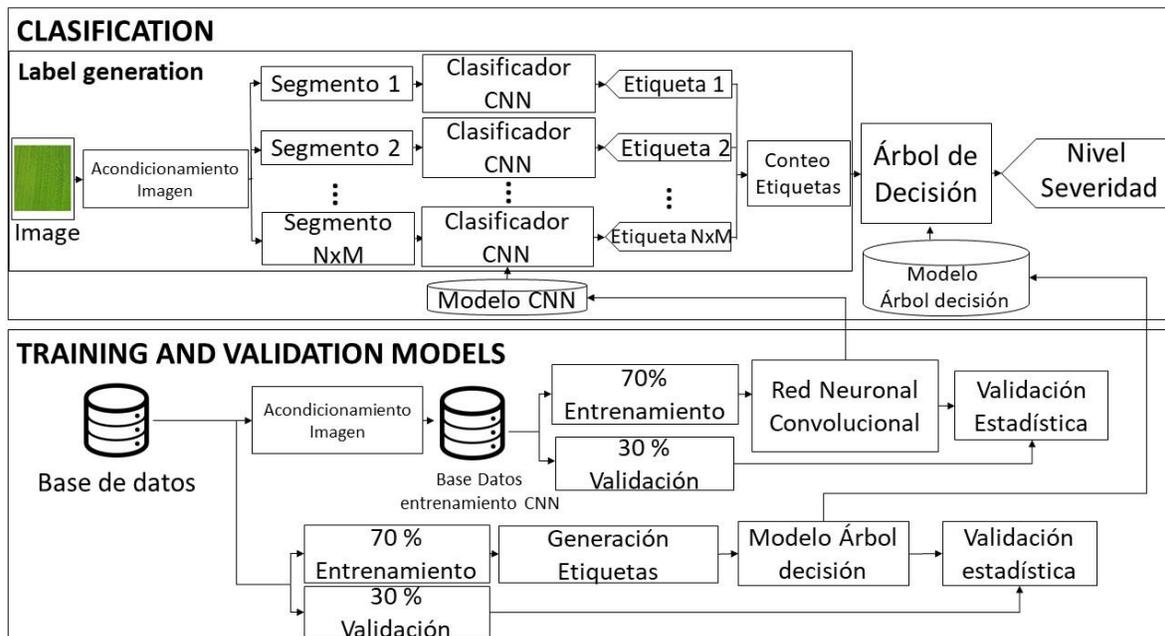


Figura 9. Metodología de aprendizaje profundo.

3.2.1 Acondicionamiento de imagen para clasificación

Esta etapa es necesaria debido a que los enfoques de clasificación con redes neuronales convolucionales requieren imágenes con resoluciones aproximadas de 96x96 píxeles [57]. Por lo anterior, se segmenta la señal en pequeñas ventanas, que puedan ser clasificadas por modelos de aprendizaje profundo. Más aún cuando las capturas hechas para la construcción de la base de datos son de alta resolución. En la figura 10 se observa el procedimiento propuesto para acondicionar una imagen. Este proceso segmenta la captura en ventanas de 500x500 píxeles y después la redimensiona a un tamaño 96x96 píxeles. Se elige este tamaño de ventana para la segmentación ya que permite conservar la información del objeto de estudio.

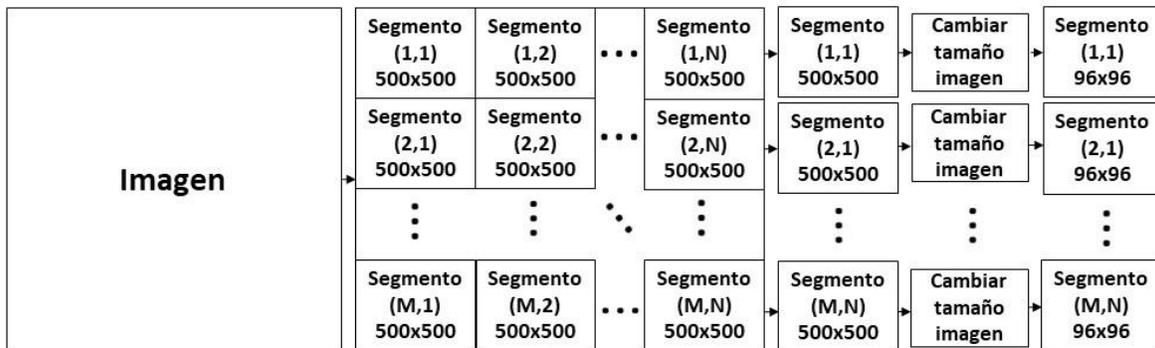


Figura 10. Diagrama bloques acondicionamiento de imagen para clasificación usando aprendizaje profundo.

3.2.2 Modelo Red Neuronal Convolutiva

El clasificador propuesto utiliza la topología de red neuronal convolutiva LENET [58], la cual se compone de una etapa de extracción de características y otra de clasificación. La extracción de características se realiza a través de capas de convolución. Estas aplican filtros en forma de ventana deslizante que computan el producto punto entre el kernel y la imagen, dando como resultado un mapa de

activación bidimensional que extrae patrones morfológicos como curvas y bordes. Esta información es vital para la descripción de la enfermedad.

Las capas de convolución 1 y 3 utilizan tamaño de kernel de 5x5 y 3x3. Además, poseen funciones de activación Relu que introducen una no linealidad a la red generando una mayor separabilidad [59]. En (1) se observa el cálculo de convolución.

$$Y_i = g\left(b_i + \sum_i K_{ij} * Y_i\right) \quad (27)$$

Donde Y_j es la salida de una neurona, j es una matriz que corresponde al producto punto entre la imagen Y_i y el kernel de convolución K_{ij} . El termino b_i corresponde al parámetro de bias de cada neurona y g es la función de activación. Las capas 2 y 4 realizan un proceso de submuestreo a partir del método MaxPooling [ref], lo que reduce el tamaño de las imágenes y genera invariancia ante posibles rotaciones y traslaciones de la entrada.

Por último, se aplanan las imágenes obteniendo un vector de características que es clasificado a través de capas totalmente conectadas tal y como se observa en la figura 11. Estas capas de conexión son numeradas de 5 a la 8 y poseen funciones de activación tanh para aumentar la separación de la ponderación de la neurona. La última capa utiliza una función de activación SoftMax [60], que calcula la probabilidad de pertenencia a la clase ("Alta", "Media", "Baja", "Sana").

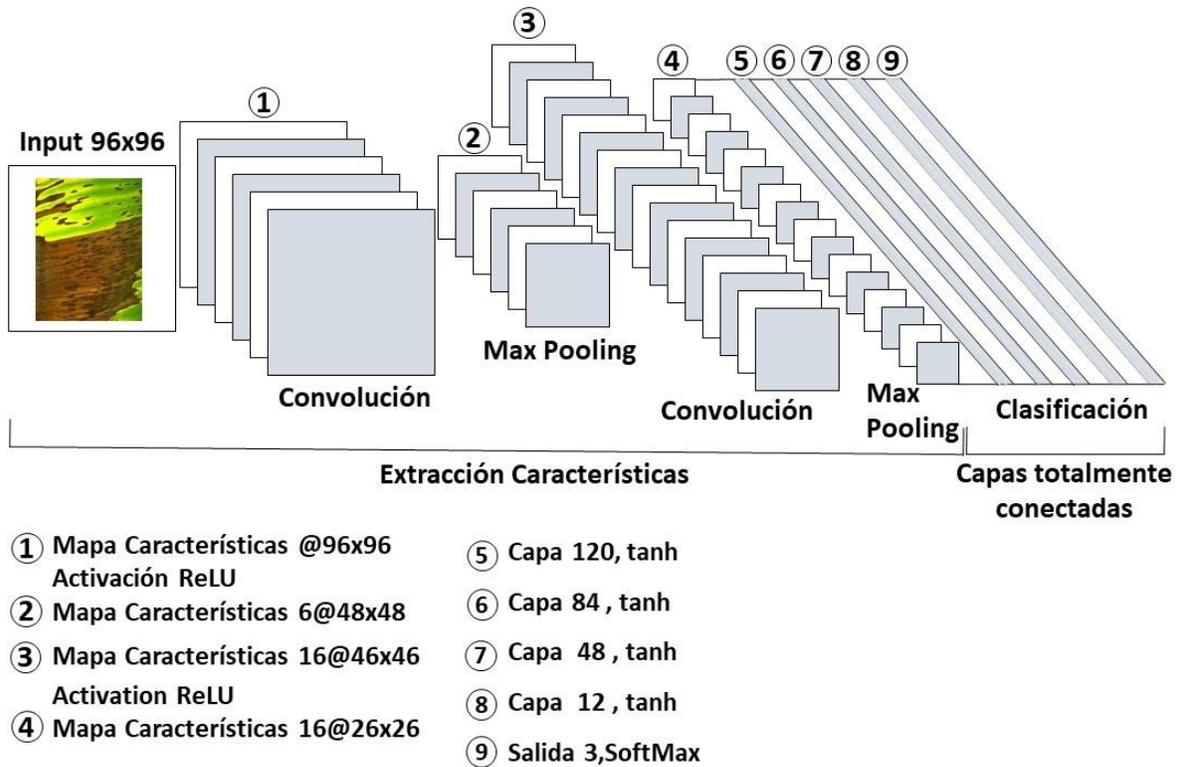


Figura 11. Topología Lenet.

3.2.3 Generación, conteo y clasificación de etiquetas

En esta etapa se clasifica cada segmento de imagen utilizando la red neuronal convolucional, esto genera un conjunto de $N \times M$ etiquetas correspondientes a los niveles de severidad "Alto", "Medio", "Bajo". El conjunto representa una codificación propia de cada estadio de enfermedad; la cual es caracterizada a través de un conteo por clase. Estas cantidades se organizan en un vector $X_{1 \times 3}$, que generan un patrón clasificable por un árbol de decisión.

En la figura 12 se observa el algoritmo utilizado para obtener el nivel de severidad de la hoja de plátano basados en el conteo de las clasificaciones dadas por la red neuronal convolucional

```

// Label Counting and Classification
Input: NxM image segments
Output: Severity level
Start
  For i = 1 until i ≤ NxM
    Etiqueta(i) = CNN(Segment(i))
    If Label(i) == "High"
      CounterHigh = CounterHigh + 1
    If Label(i) == "Medium"
      CounterMedium = CounterMedium + 1
    If Label(i) == "Low"
      CounterLow = CounterLow + 1
  End for
  x = [ CounterHigh, CounterMedium, CounterLow ]
  Severity level = DT(x)
End
// CNN() = Convolutional neural network classifier
// DT() = Decision Tree classifier

```

Figura 12. Algoritmo Conteo de etiquetas.

3.2.4 Entrenamiento de Modelos de clasificación

El entrenamiento de la CNN se realiza con la base de datos de imágenes reducidas que se presentará en la siguiente sección. Como se muestra en la tabla 1, la red LENET propuesta aprende los pesos y sesgos usando el algoritmo Adadelta, sigue una función de perdida Categorical crossentropy y evalúa el modelo siguiendo la métrica Accuracy.

Parameter	Function loss	Optimization algorithm	Metrics	Batch Size	Iterations
Value	Categorical crossentropy	Adadelta	Accuracy	12	10

Tabla 1. Parámetros de entrenamiento Red Neuronal Convolutacional.

3.2.5 Entrenamiento del Árbol de decisión

El entrenamiento del árbol de decisión se realiza con la base de datos que se presentará en la siguiente sección. el modelo usado, se entrena con 4 nodos, utilizando el método de optimización Gini's diversity index sin criterio de poda.

3.2.6 Base de datos para entrenamiento red neuronal

Las muestras que posee la base de datos poseen una resolución (4128x2322) y 13Mp (3120x4160), con esta resolución es inviable entrenar los parámetros de la red debido a que conllevaría un costo computacional demasiado alto. Es por esto que a cada imagen de la base de datos descrita en la sección 3.1.1 se le aplica el proceso de acondicionamiento explicado en la sección 3.2.1 y se obtienen subimágenes con un tamaño 96x96 pixeles. Luego estas imágenes son depuradas y etiquetas según la escala de Fouré. Esto permite obtener 4.244 muestras. En la figura 13, se observa algunos ejemplos de la nueva base de datos y la población por clase.

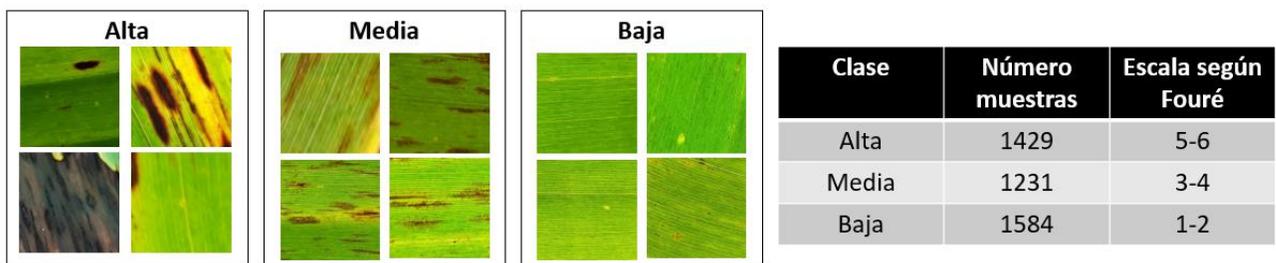


Figura 13. Base de datos para el Entrenamiento de la CNN.

Métodos de prueba

La metodología propuesta contempla las pruebas necesarias para validar cada una de las etapas descritas en la figura 1, a continuación, se explica el procedimiento seguido para la correspondiente validación.

Con el fin de determinar qué sistema de representación genera una mayor separabilidad en la etapa de acondicionamiento de imágenes se comparan los modelos de color más usados en la literatura. Para esto se realiza una inspección manual de 100 muestras por cada clase y se elige el espacio de color que mejor segmenta la enfermedad.

En la etapa de extracción de descriptores se valida el número de bins necesarios para que el descriptor H aporte mayor información para la clasificación la enfermedad requiriendo el menor tiempo de computo posible, esto se logra realizando un estudio donde se observa la eficiencia lograda para diferentes números de bins y el costo computacional requerido para validar el modelo.

El análisis de dimensionalidad parte de calcular la correlación de la matriz de características, esto genera un método de visualización que magnifica la dependencia entre las características extraídas. El cálculo de la correlación revela una dependencia directamente proporcional en los valores cercanos a 1 (color amarillo), dependencia inversamente proporcional en los valores cercanos a -1 (color azul oscuro).

Posteriormente se valida el número de vecinos (k) que requiere el método Relieff para que los rangos y pesos computados generen una correcta representación de los datos. Esto se logra variando el valor de k y observando la eficiencia de clasificación obtenida

Como se explicó en la metodología se realiza un análisis de dimensionalidad a nivel de característica y de descriptor a partir de tres pruebas, donde se determina que proceso genera una eficiencia adecuada requiriendo el menor tiempo de cómputo. En la primera prueba se calcula el aporte de información de cada característica según lo obtenido por Relief. En la segunda prueba se suma el peso de las características pertenecientes a cada descriptor, se ordenan mayor a menor según su peso y se calcula el aporte de información de cada descriptor. La tercera prueba se realiza siguiendo el criterio de los autores [54]. Para comparar el costo computacional se mide el tiempo que tarda el computador en calcular los descriptores en las tres pruebas propuestas, además se mide el tiempo necesario para realizar el entrenamiento del modelo de clasificación.

Se valida la elección del modelo de clasificación para el conjunto de entrenamiento X completo y reducido, comparando los resultados de un modelo simple respecto a un clasificador compuesto por dos modelos jerárquicos. Se coteja el clasificador propuesto respecto a otros clasificadores usados en el estado del arte, para validar el aporte en la reducción de dimensiones, se realizan pruebas de entrenamiento con el conjunto de descriptores completo y con la reducción propuesta en la tercera prueba. Todas las pruebas de clasificación se validan usando cross-validation con 100 folds donde se fraccionan las muestras de forma aleatoria siguiendo una distribución gaussiana.

Finalmente se compara la metodología de clasificación usando aprendizaje de máquina tradicional respecto a una propuesta metodológica usando aprendizaje profundo.

En este trabajo se utiliza el 70% de los datos para entrenamiento y el restante para validación. Todas las pruebas se realizan en un computador con un procesador Intel Core i7 5500U y 8Gb de memoria RAM.

4. ANÁLISIS Y RESULTADOS

La comparación de los espacios de color revela que el mejor resultado se obtiene usando el canal V de HSV, debido a que logra segmentar la mayor cantidad de pixeles enfermos como se observa en la figura 14.

De la tabla 2 se determina se determina que 15 bins es el parámetro adecuado para computar el descriptor H debido a que requiere el menor tiempo de computo manteniendo una eficiencia similar a las otras pruebas realizadas.

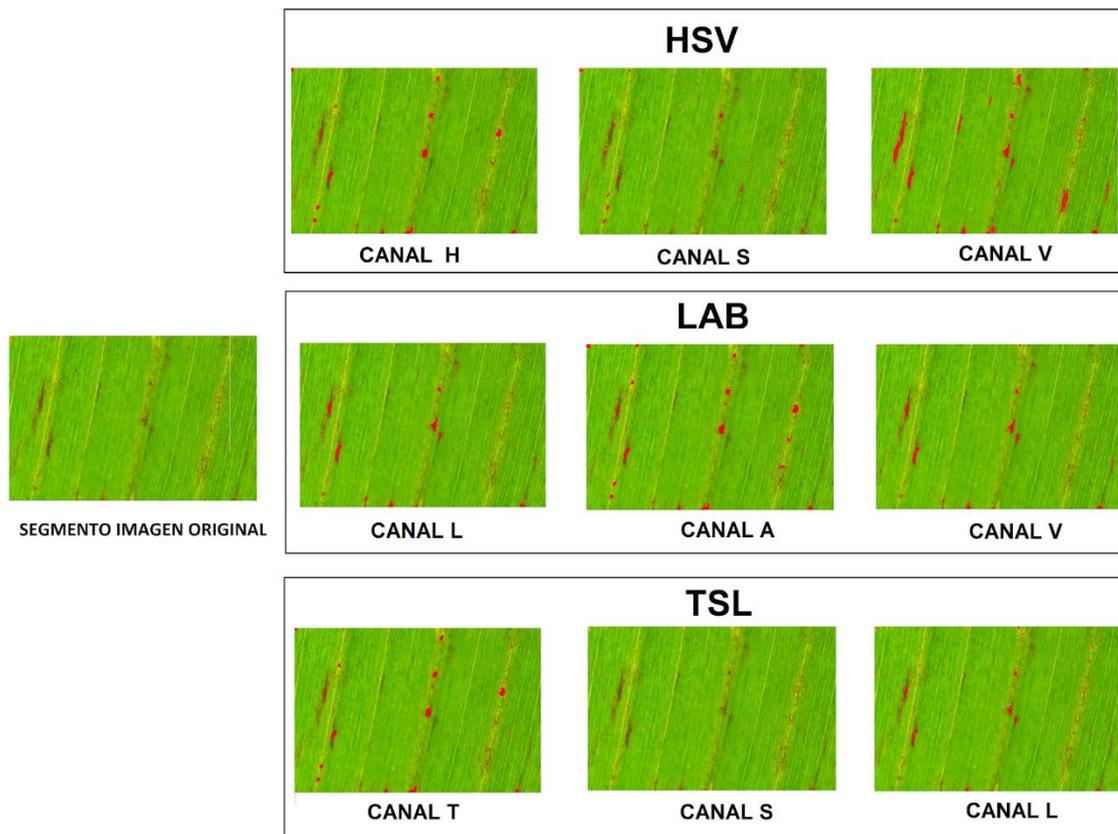


Figura 14. Prueba espacios de color

Numero de Bins	Eficiencia [%]	Tiempo [s]
7	82.17 ±1.43	490.25±6.21
15	84.17 ± 1.12	662.25 ± 8.93
20	84.53± 1.21	880.79±9.34
25	84.56± 1.19	997.67±9.73
30	84.52± 1.15	1021.35 ± 9.94

Tabla 2. Eficiencia de clasificación vs número de bins.

La gráfica de correlación (Figura 15) revela que existe una alta correlación entre algunas características de la matriz X completa evidenciando la necesidad de un análisis de los datos obtenidos. Además, se observa que el cálculo de algunas características genera valores indeterminados o cero, lo que podría incurrir en un mal entrenamiento de los modelos de aprendizaje de máquina, por lo tanto, las muestras que obtienen estos valores se eliminan.

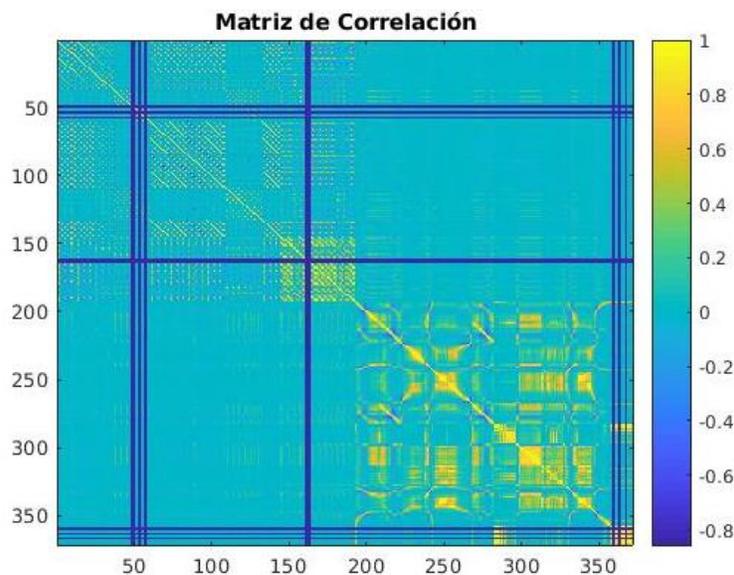


Figura 15. Correlación matriz de características.

La figura 16 revela que la prueba para la elección del número de vecinos de Relieff no es concluyente, sin embargo, se evidencia que con 140 características es posible lograr una eficiencia muy cercana a la obtenida con el conjunto de datos completo. Por tanto, en la tabla 3, se presenta la eficiencia en términos de media y desviación estándar para 140 características, aunque no existen diferencias significativas para las variaciones de k. Se elige k=7 ya que con este valor se obtiene la desviación estándar más baja.

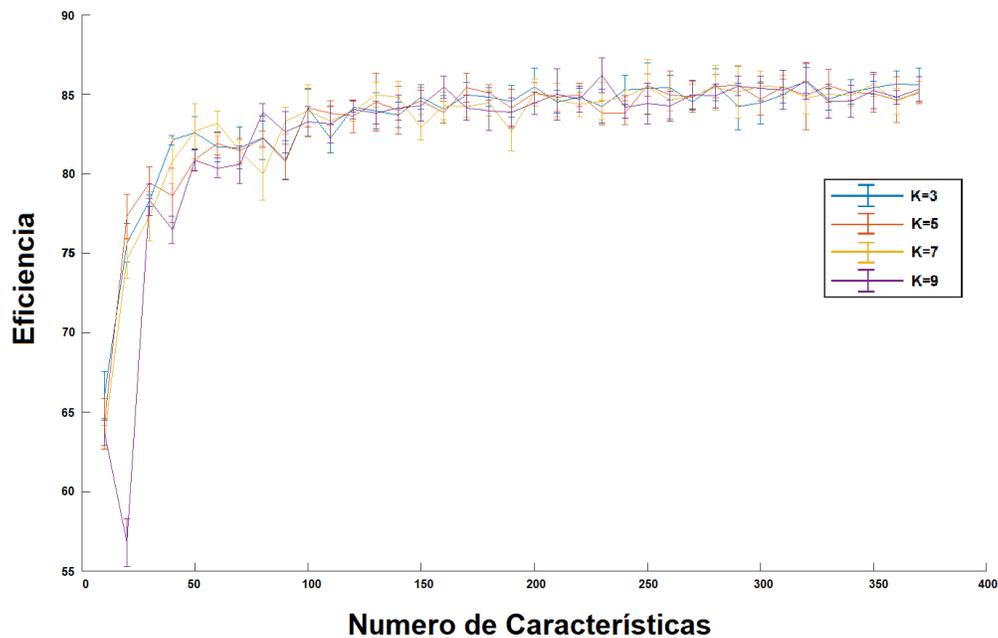


Figura 16. Eficiencia vs número de características para K vecinos

K Vecinos	3	5	7	9
Eficiencia (%) con 140 Características según Relieff	83.88 ± 1.25	84.12 ± 1.31	84.15 ± 1.17	84.03 ± 1.23

Tabla 3. Eficiencia vs cantidad de vecinos.

El análisis de dimensionalidad inicia con una prueba donde se calcula el peso de cada característica según la ponderación computada por Relieff. Se verifica la eficiencia

de clasificación respecto al número de características usadas para el entrenamiento. De la gráfica además se observa que una eficiencia del $81.47\% \pm 1.02\%$ se logra entrenando el modelo con 53 características como se observa en la figura 17 (a), sin embargo, para obtener esta cantidad de características se requiere computar 25 descriptores por tanto la prueba no conlleva a una reducción significativa del costo computacional.

A partir de la figura 17 (b), se determina que es posible construir la matriz X reducida usando los 17 vectores con más peso, estos contienen 111 características y generan una eficiencia del $81.08\% \pm 1.09\%$. Aunque el desempeño es menor al obtenido utilizando el análisis por característica en un 0.39%, el costo computacional se reduce como se muestra en la tabla 4.

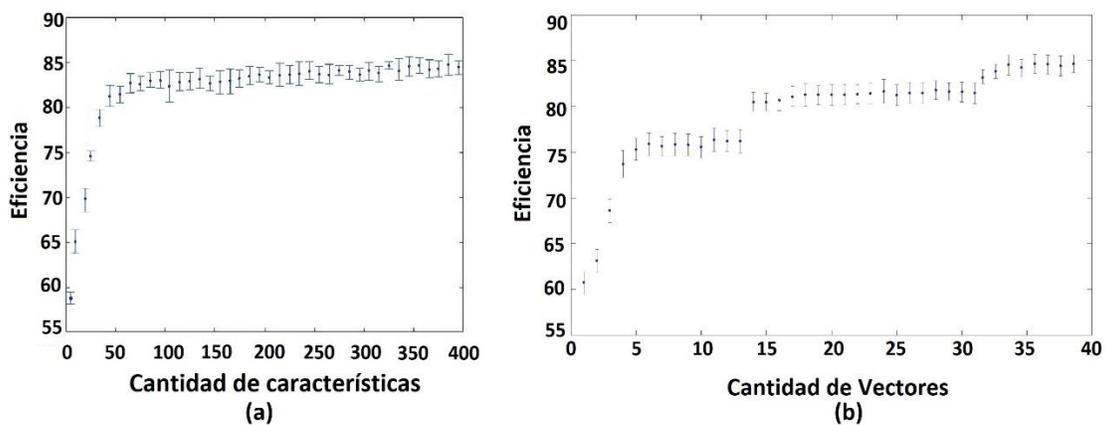


Figura 17. Análisis de dimensionalidad a nivel de características (a) y a nivel de descriptores (b).

La tercera prueba se realiza siguiendo el criterio de los autores [54], Este conjunto de datos genera una eficiencia del $72.49\% \pm 1.46\%$. Aunque los resultados en términos de eficiencia son bajos, el costo computacional se reduce considerablemente. Por lo que se propone al conjunto de datos anterior agregar el

descriptor H para los canales HSV. Este criterio se tiene cuenta debido a que este descriptor presenta un alto peso según las pruebas realizadas con Relieff. Esta adición genera un espacio de representación de 61 características y garantiza una eficiencia del $80.58\% \pm 1.27\%$, esto supone una eficiencia competitiva con respecto a las dos pruebas anteriores y el costo computacional se reduce un 92.36% respecto a la prueba 1 y un 86.84% respecto a la prueba 2, como se observa en la tabla 4.

	Eficiencia	Tiempo Extracción de Descriptores para una muestra [s]	Tiempo 1 iteración del modelo de aprendizaje [s]
Prueba 1	$81.47 \pm 1.02\%$	96.76 ± 1.39	2.82 ± 0.31
Prueba 2	$81.08 \pm 1.09\%$	56.16 ± 1.50	4.84 ± 0.53
Prueba 3	$80.58 \pm 1.27\%$	7.39 ± 1.39	2.28 ± 0.26

Tabla 4. Comparación de tiempos de las pruebas realizadas.

Obtenidas las matrices de representación, se válida la propuesta de un clasificador jerárquico entrenado con clases balanceadas. En la tabla 5 se presentan las pruebas realizadas con el modelo Ensemble Boosted Tree y en la tabla 6 Ensemble Boosted Tree jerárquico. Se observa que al usar un modelo jerárquico se logra aumentar la eficiencia en la clasificación y se eleva considerablemente el porcentaje de acierto para la clase Sana de un $56.39\% \pm 5.32\%$ a 90.94 ± 2.30 . Las clases Alta y Media no presentan cambios significativos, sin embargo, se genera una reducción en la clase Baja, aunque esta no es relevante teniendo en cuenta las mejoras presentadas. Lo anterior se debe a este método garantiza un entrenamiento balanceado.

	Alta	Media	Baja	Sana
Alta	95,90±0,99%	3,57±1,01%	0,52±0,40%	0
Media	4,97±1,62%	80,40±2,50%	14,62±2,61%	0
Baja	0,02±0,10%	5,70±1,36%	88,68±1,72%	5,59±1,42%
Sana	0	0,01±0,13%	43,58±5,32%	56,39±5,32%

Tabla 5. Matriz de confusión clasificador Ensemble Boosted Tree simple, entrenado con X completo.

	Alta	Media	Baja	Sana
Alta	97,04±1,15%	2,71±1,08%	0,20±0,22%	0,03±0,94%
Media	9,93±2,59%	82,03±3,01%	7,96±1,97%	0,06±0,17%
Baja	0,25±0,48%	6,93±1,86%	75,52±2,87%	17,28±2,21%
Sana	0,35±0,67%	0,05±0,23%	8,64±2,20%	90,94±2,30%

Tabla 6. Matriz de confusión clasificador Ensemble Boosted Tree jerárquico, entrenado con X completa.

En la tabla 7 se presentan las diagonales principales de cada clasificador y a su vez el tiempo de cómputo necesario para validar cada modelo usando el conjunto de datos completo. En esta se observa que el método propuesto presenta una mayor eficiencia en las clases Alta y Sana, sin embargo, es superado en las clases Baja por la SVM además de presentar el mayor tiempo necesario para el entrenamiento del modelo.

	Ensemble Boosted Tree Jerárquico	SVM	Decision Tree	KNN
Alta	97.04±1.15%	90.22±1.88%	94.86±1.47%	74.73±2.58%
Media	82.03±3.01%	84.25±2.62%	72.00±4.54%	52.78±3.69%
Baja	75.52±2.87%	85,20±1.89%	81.53±4.06%	89.13±1.85%
Sana	90.94±2.30%	55,30±4.89%	54.28±11.37%	18.05±3.53%
Total	84.17 ± 1.12%	81.18±0.93%	81.15±1.20%	68.7±1.31%
Tiempo [s]	662.25±6.93	100,79±2.53	40.70±2.26	67.29±2.26

Tabla 7. Comparación diagonal principal matrices de confusión de los modelos de clasificación usando el conjunto de datos completo.

Para ampliar la información presentada al lector y aunque ya se realizó la comparación de los clasificadores, en las tablas 8, 9 y 10 se describen las matrices de confusión resultados de la validación de los modelos SVM, árbol de decisión y KNN respectivamente.

	Alta	Media	Baja	Sana
Alta	90,22±1,88%	9,40±1,88%	0,30±0,31%	0,06±0,3%
Media	5,24±1,47%	84,25±2,62%	10,43±2,46%	0,06±0,20%
Baja	0,13±0,19%	8,56±1,19%	85,20±1,89%	6,09±1,51%
Sana	0%	0,22±0,43%	44,47±48,3%	55,30±4,89%

Tabla 8. Matriz de confusión clasificador SVM, entrenado con X completa.

	Alta	Media	Baja	Sana
Alta	94,86±1,47%	4,39±1,37%	0,72±0,48%	0,01±0,06%
Media	5,95±2,21%	72,00±4,54%	21,64±4,31%	0,40±0,43%
Baja	0,09±0,18%	7,92±2,43%	81,53±4,06%	10,45±3,74%
Sana	0,01±0,13%	0,16±0,42%	45,52±11,32%	54,28±11,37%

Tabla 9. Matriz de confusión clasificador Árbol de decisión, entrenado con X completa.

	Alta	Media	Baja	Sana
Alta	74,73±2,58%	13,82±2,03%	11,15±1,73%	0,28±0,29%
Media	8,01±2,10%	52,78±3,69%	38,67±3,62%	0,52±0,46%
Baja	0,98±0,53%	6,46±1,49%	89,13±1,85%	3,40±1,14%
Sana	0,85±0,81%	1,35±1,03%	80,00±3,74%	18,05±3,53%

Tabla 10. Matriz de confusión clasificador KNN, entrenado con X completa.

En la tabla 11 se evidencia que la metodología propuesta en este trabajo presenta una eficiencia mayor en comparación con las otras propuestas de clasificación con el conjunto de datos reducidos además de presentar una reducción considerable del tiempo de computo requerido. No obstante, la clase Baja sigue siendo superado por otros clasificadores, aunque esto se compensa con la eficiencia alcanzada en la clase “Sana” (78.54±4.52%) que supera ampliamente los resultados obtenidos por los otros clasificadores. En las tablas 12,13,14 y 15, se presentan las matrices de confusión de los modelos Ensemble Boosted Tree, SVM, árbol de decisión y KNN respectivamente.

	Ensemble Boosted Tree Jerárquico	SVM	Arbol Decisión	KNN
Alta	95.41±1.36%	80.20±2.35%	90,59±1.79%	76.36±2.33%
Media	74.47±3.37%	71.68±3.01%	68,27±5.20%	55.00±3.24%
Baja	72.47±2.83%	78.61±2.36%	84,95±3.92%	86.92±1.76%
Sana	78.54±4.52%	36,33±5,20%	36,03±7.8%	18.63±3.29%
Total	80.58±1.05%	73.38±1.33%	78.02±1.22%	69.06±1.18%
Tiempo[s]	245±3.29	48.09±1.21	15.71±.64	16.34±0.53

Tabla 11. Comparación diagonal principal matrices de confusión de los modelos de clasificación usando el conjunto de datos reducido.

	Alta	Media	Baja	Sana
Alta	95,41±1,36	3,89±1,27	0,62±0,37	0,06±0,13
Media	14,62±2,96	74,47±3,37	10,12±2,49	0,77±0,52
Baja	0,58±0,49	7,37±1,59	72,47±2,83	19,54±2,22
Sana	0,56±0,56	0,41±0,79	20,47±4,03	78,54±4,25

Tabla 12. Matriz de confusión clasificador Ensemble Boosted Tree, entrenado con X reducida.

	Alta	Media	Baja	Sana
Alta	80,20±2,35	17,29±2,23	2,08±0,81	0,41±0,36
Media	13,47±2,51	71,68±3,01	14,11±2,02	0,71±0,55
Baja	1,83±0,60	11,05±1,34	78,61±2,36	8,4±1,80
Sana	1,83±1,34	1,13±0,96	60,69±5,42	36,33±5,20

Tabla 13. Matriz de confusión clasificador SVM, entrenado con X reducida.

	Alta	Media	Baja	Sana
Alta	90,59±1,79	7,81±1,67	1,43±0,88	0,15±0,20
Media	10,24±2,56	68,27±5,20	20,78±4,95	0,69±0,68
Baja	0,35±0,40	8,00±2,59	84,95±3,92	6,68±2,79
Sana	0,01±0,09	0,37±0,99	63,58±7,87	36,03±7,80

Tabla 14. Matriz de confusión clasificador Árbol de decisión, entrenado con X reducida.

	Alta	Media	Baja	Sana
Alta	76,36±2,33	18,45±2,16	4,9±1,15	0,23±0,26
Media	24,82±3,10	55,00±3,24	19,54±1,76	0,62±0,52
Baja	3,53±1,00	5,31±1,21	86,92±1,76	4,22±1,10
Sana	2,83±1,66	1,11±0,97	77,41±3,42	18,63±3,29

Tabla 15. Matriz de confusión clasificador KNN, entrenado con X reducida.

Mostrados los resultados obtenidos con los métodos de aprendizaje de máquina tradicional, se presentan los resultados obtenidos con la metodología de aprendizaje profundo.

Inicialmente se presenta la matriz de confusión producto del clasificador LENET con tres clases, “Alta”, “Media”, “Baja” obteniendo una eficiencia del 90.23% ± 1.85%. Como se presenta en la tabla 16, se observan resultados competitivos respecto a la metodología que hace uso del método Ensemble Boosted Tree.

	Alta	Media	Baja
Alta	87.16 ± 2.12%	12.53 ± 1.98%	1.41 ± 0.84%
Media	9.22 ± 1.66%	80.97 ± 1.91%	6.49 ± 2.45%
Baja	0.48 ± 0.56%	0.54 ± 0.71%	99.18 ± 1.74%

Tabla 16. Matriz de confusión Red Neuronal Convolutiva Lenet con 3 clases.

Posteriormente, en la tabla 17 se presentan los resultados del clasificador con árbol de decisión, que corresponde a las etiquetas generada por la metodología completa obteniendo una eficiencia del $89.4\% \pm 1.47\%$.

	Alta	Media	Baja
Alta	$91\% \pm 1.92\%$	$8\% \pm 1.02\%$	$2\% \pm 0.75\%$
Media	$11\% \pm 1.21\%$	$71\% \pm 2.33\%$	$19\% \pm 1.31\%$
Baja	$<1\% \pm 0.54\%$	$4\% \pm 0.91\%$	$96\% \pm 2.32\%$

Tabla 17. Matriz de confusión Árbol de decisión, resultado de la metodología con aprendizaje profundo.

Sin embargo, es necesario saber cuándo una planta se encuentra en ausencia de la enfermedad, ya que esto brinda mayor información al campesino, por tanto, se incluye la clase sana como se presenta en la tabla 18. Esto disminuye la eficiencia a un $72.05\% \pm 1.7\%$ principalmente por el traslape entre la clase baja y sana. Posteriormente, en la tabla 19 se presenta la matriz de confusión del clasificador árbol de decisión que obtiene una eficiencia del $74.6\% \pm 1.7$ y que genera las etiquetas de la metodología completa, se observa la imposibilidad del clasificador para separar entre las clases baja y sana, debido al traslape que ya se presentaba en las etiquetas generadas por la Red Neuronal Convolutiva

	Alta	Media	Baja	Sana
Alta	90.31 ± 2.3	9.76 ± 1.8	0.45 ± 0.31	0.39 ± 0.3
Media	6.77 ± 1.53	85.08 ± 1.4	11.31 ± 1.30	1.99 ± 1.11
Baja	0.24 ± 0.26	5.91 ± 1.25	80.54 ± 2.31	7.56 ± 1.8
Sana	0 ± 0	2.31 ± 1.13	93.21 ± 2.14	14.34 ± 1.35

Tabla 18. Matriz de confusión Red Neuronal Convolutiva Lenet con 4 clases.

	Alta	Media	Baja	Sana
Alta	90±2.12	7±1.35	3±0.97	0±0
Media	12±1.24	57±3.03	31±2.49	0±0
Baja	1±0.41	7±1.43	93±1.94	0±0
Sana	0±0	1±0.42	99±0.60	0±

Tabla 19. Matriz de confusión Árbol de decisión, resultado de la metodología con aprendizaje profundo con 4 clases.

La red neuronal convolucional LENET presenta resultados competitivos (ver tablas 16 y 17) en la tarea de clasificación de tres clases, obteniendo $87.16 \pm 2.12\%$, $80.97 \pm 1.91\%$ y $99.18 \pm 1.74\%$, en las clases “Alta”, “Media” y “Sana” respectivamente sin embargo, cuando se desea clasificar 4 clases como en la metodología de aprendizaje de máquina tradicional propuesta en este documento, se disminuye la eficiencia considerablemente, ya que esta red no genera una correcta separabilidad entre algunas clases, lo que si se logra con la propuesta de un clasificador jerárquico usando el modelo Ensemble Boosted Tree (ver tablas 6 y 12).

En términos de costo computacional, el modelo LENET tarda $156s \pm 1.2s$ en aprender los parámetros que generan la eficiencia mostrada, lo que supone un menor tiempo respecto al modelo Ensemble Boosted Tree que tarda $245s \pm 3.29s$.

5. CONCLUSIONES

En esta sección se presentan las conclusiones del trabajo realizado, de manera general se puede concluir que:

- La metodología propuesta permite clasificar estadios de la enfermedad desde sus primeras etapas, lo que permitiría a un agricultor efectuar una acción

correctiva, mejorando la calidad del producto y evitando pérdidas económicas.

También se realiza una comparación frente a metodologías de aprendizaje de máquina tradicional respecto a técnicas de aprendizaje profundo.

5.1 Metodología aprendizaje de máquina clásico

- Se estudia el aporte de información de las transformadas FFT, DWT e Histogramas, para los espacios de representación HSV, TSL, LAB, RGB y GRAY a partir de los pesos calculados en tres pruebas como se observa en la sección de resultados. Se comprueba que el mayor aporte se presenta en el espacio Gray, debido a que los cambios de textura que produce la enfermedad y que se detectan usando transformadas espacio frecuencia invariantes al color, sin embargo, se añaden los Histogramas del canal HSV, lo que permite lograr una eficiencia esperada.
- La técnica de reducción de dimensionalidad relieff es un punto inicial para determinar el aporte de cada descriptor, sin embargo, no solo se deben organizar los descriptores por peso, si no que se debe validar el aporte de las diferentes combinaciones de descriptores. Lo anterior se verifica en la tercera prueba donde se agrega el descriptor FFTGRAT que no se encuentra entre los mejor puntuados por Relieff, pero permite mantener una eficiencia competitiva con una menor cantidad de descriptores.
- El análisis de correlación y las pruebas de dimensionalidad evidencian la importancia de un estudio a la base de datos, ya que existen características que no generan información relevante e incrementan el tiempo de computo necesario para procesar el método propuesto.

- En la etapa de procesamiento de imágenes el espacio HSV genera una mayor separabilidad determinar entre pixeles sanos y enfermos.
- El método propuesto requiere mayor costo computacional, debido a la necesidad de utilizar dos clasificadores jerárquicos, no obstante, permite mantener una eficiencia esperada en los primeros estadios de la enfermedad, lo que no se logra utilizando otras técnicas de clasificación propuestas en el estado del arte. Detectar la enfermedad en los primeros estadios es de vital importancia, ya que permite al agricultor tomar una acción preventiva, evitando pérdidas económicas.
- La base de datos se encuentra desbalanceada debido a la naturaleza de la enfermedad, esto requiere el uso de métodos como el de particionar los datos para evitar clasificadores sesgados. Además, evidencia la fuerte incidencia que presenta la Sigatoka Negra en los cultivos productivos del departamento de Risaralda.

5.2 Metodología usando aprendizaje profundo

- Dividir la imagen en segmentos permite abordar el problema de clasificación a partir de redes neuronales convolucionales, dado que las piscas que produce la enfermedad son de tamaño menor a 1 cm^2 no es posible reducir el tamaño de la imagen significativamente debido a que se perdería la información para la clasificación del patrón.
- Las redes neuronales convolucionales resultan ser más eficientes en la clasificación de patrones definidos, como rostros u objetos, sin embargo, la Sigatoka Negra presenta patrones aleatorios, por tanto, descriptores en conjunto como la transformada Wavelet, Fourier e histogramas aportan mayor información que los procesos de convolución.

6. TRABAJOS FUTUROS

A continuación, se presentan los trabajos futuros que se plantean después de realizar este documento.

- Probar otro conjunto de descriptos de textura como SIFT y SURF que podrían entregar mayor información para la clasificación.
- Generar un modelo ensamblado que no solo incluya un único modelo simple, sino también una fusión de diferentes clasificadores.
- Generar un algoritmo de ventana deslizante que recorra la imagen identificando los sectores enfermos en lugar de generar ventanas estáticas para la metodología con aprendizaje profundo.
- Desarrollar un algoritmo no solo de clasificación sino también que ubique los píxeles enfermos, con el objetivo de presentar mayor información al agricultor.

7. REFERENCIAS

- [1] F. Jiménez, «El Colombiano,» 04 01 2016. [En línea]. Available: <https://www.elcolombiano.com/negocios/inventario-agropecuario-de-colombia-NN3384078>. [Último acceso: 2019 02 6].
- [2] C. Espinal, «Asohofrucol,» [En línea]. Available: <http://www.asohofrucol.com.co/archivos/cadenas/platano.pdf>. [Último acceso: 05 02 2019].
- [3] A. Rodriguez y J. Rodriguez, Aspectos Socioeconómicos del cultivo de Plátano en Colombia, Armenia.
- [4] Mintic, «Datos Gobierno de colombia,» 2019. [En línea]. Available: <https://www.datos.gov.co/es/Agricultura-y-Desarrollo-Rural/Cadena-Productiva-Pl-tano-Area-Producci-n-Y-Rendim/uw9z-pgwy>. [Último acceso: 05 02 2019].

- [5] «Tecnología aplicada a la productividad del campo colombiano,» *Dinero*, 19 01 2016.
- [6] «Las tecnologías que podrán ‘salvar’ el agro colombiano,» *Portafolio*, 12 02 2019.
- [7] J. Cuevas, *Sistema de adquisición y Análisis de Información para el Diagnóstico de Sigatoka Negra*, Universidad del Colima, 2015.
- [8] E. Vargas, «RepositorioTec,» *Monitoreo asistido de plantaciones agrícolas: Técnicas de procesamiento digital de imágenes e inteligencia artificial para diagnosticar y controlar la incidencia de la Sigatoka Negra en plantaciones de banano y plátano curraré*, 2008.
- [9] E. Álvarez, A. Pantoja, L. Gañán y G. Ceballos, «FAO,» [En línea]. Available: <http://www.fao.org/3/as089s/as089s.pdf>. [Último acceso: 15 02 2019].
- [10] G. Cedeño, C. Suarez, D. Vera y C. Fadda, «Early detection of resistance to *Mycosphaerella fijiensis* in local genotypes of *Musa* in Ecuador,» *Scientia Agropecuaria*, vol. 8, pp. 29-42, 2017.
- [11] R. Ghini, E. Hamada y R. Goncalves, «Análise de risco das mudancas climáticas globais sobre a sigatoka negra,» *Embrapa Meio Ambiente-Artigo em periódico indexado (ALICE)*, 2007.
- [12] B. Ribadeneira y . O. Ruiz, «Desarrollo de un modelo porcentual para la evaluación de sigatoka negra en condiciones de invernadero,» *Espol, Guayaquil*, 2016.
- [13] J. Freitez, M. Ablan y C. Gómez, «Propuesta de modelos predictivos del brote de la Sigatoka Negra para las plantaciones de plátano al sur del Lago de Maracaibo, Venezuela,» *Revista Científica UDO agrícola*, vol. 9, nº 1, pp. 191-198, 2009.
- [14] L. Calvo, M. Guzmán y J. Guzmán, «Considerations about Application of Machine Learning to the Prediction of Sigatoka Disease,» *World Conference on Computers in Agriculture and Natural Resources*, vol. 1, nº 1, 2014.
- [15] W. Morales, S. Modenese, H. Nascimento, C. Padovani, L. Castro y K. Botigeli, «Digital image processing for identification of Black Sigatoka,» *INTERNATIONAL CONFERENCE OF AGRICULTURAL ENGINEERING*, 2008.
- [16] S. Pinto, J. Escobar, K. Murcia, M. Romero y B. Valencia, «Validación mediante el Método PPI de un Algoritmo Computacional para la medición automática del área de afectación por Sigatoka negra en imágenes de hojas de plátano del Departamento del Meta, Colombia,» *SISTEMAS, CIBERNÉTICA E INFORMÁTICA*, vol. 14, nº 1, pp. 24-28, 2017.
- [17] N. Balaji, «A Survey Paper on Crop Disease Identification and Classification Using Pattern Recognition and Digital Image Processing Techniques,» *IOSR Journal of Computer Engineering (IOSR-JCE)*, pp. 14-17, 2017.
- [18] I. Singh y A. Misra, «Detection of plant leaf diseases using image segmentation and soft computing techniques,» *Information Processing in Agriculture*, vol. 4, nº 1, pp. 41-49, 2017.

- [19] T. Vipinadas, «Detection and Grading of diseases in Banana leaves using Machine Learning,» *International Journal Of Scientific & Engineering Research*, vol. 7, pp. 916-924, 2016.
- [20] A. Camargo y J. Smith, «An image-processing based algorithm to automatically identify plant disease visual symptoms,» *Biosystems engineering*, vol. 1, nº Elsevier, pp. 9-21, 2009.
- [21] «Evaluación de un método de Aprendizaje Supervisado para la detección de las enfermedades, Antracnosis y Phytophthora Infestans en cultivos de fruta de Risaralda,» *Universidad Tecnológica de Pereira*, 2017.
- [22] W. Liao, D. Ochoa, Y. Zhao, G. Villegas y W. Philips, «Banana Disease Detection by Fusion of Close Range Hyperspectral Image and High-Resolution Rgb Image,» *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, 2018.
- [23] M. Prabukumar, B. Krishna y J. Kamalakannan, «Image Processing and Pattern Classification Technique in a Machine Vision System that Identifies and Classifies the Plant Diseases Based on the Visual Symptoms,» *International Journal of Advanced Research in Computer Science*, vol. 1, nº 4, pp. 302-307, 2010.
- [24] V. Pooja, R. Das y V. Kanchana, «Identification of plant leaf diseases using image processing techniques,» *IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR)*, pp. 130-133, 2017.
- [25] S. Ramesh , P. Vinod , M. Niveditha, R. Pooja , P. Bhat y N. Shashank, «Plant Disease Detection Using Machine Learning,» *2018 International Conference on Design Innovations for 3Cs Compute Communicate Control (ICDI3C)*, 2018.
- [26] J. Amara, B. Bouaziz y A. Algergawy, «A Deep Learning-based Approach for Banana Leaf Diseases Classification,» *Lecture Notes in Informatics (LNI), Gesellschaft für Informatik*, pp. 79-88, 2017.
- [27] V. M y A. Thamizharasi, «Banana Leaf Disease Identification Technique,» *International Journal of Advanced Engineering Research and Science*, vol. 3, nº 6, pp. 120-124, 2016.
- [28] Promusa, 13 09 2016. [En línea]. Available: <http://www.promusa.org/Sistema+de+predicci%C3%B3n+biol%C3%B3gica+para+controlar+la+Sigatoka+negra>. [Último acceso: 5 03 2019].
- [29] S. Modenese, «Processamento digital de imagens para identificação da sigatoka negra em bananais utilizando análise de componentes principais e redes neurais artificiais,» 2008.
- [30] D. Prieto y F. Moreno, «Análisis y monitoreo remoto de las variables ambientales en un entorno indoor, por medio de un dispositivo móvil con android,» *Ciencia y Poder aéreo*, vol. 10, nº 1, pp. 187-194, 2015.

- [31] P. Manjunatha , A. Verma y A. Srividya, «Multi-Sensor Data Fusion in Cluster based Wireless Sensor Networks Using Fuzzy Logic Method,» *2008 IEEE Region 10 and the Third International Conference on Industrial and Information Systems*, pp. 1-6.
- [32] J. Rhee, J. Im y G. Carbone, «Monitoring agricultural drought for arid and humid regions using multi-sensor remote sensing data,» *Elsevier*, vol. 114, nº 12, pp. 2875-2887, 2010.
- [33] J. Hernández, N. Sanabria, G. Blanco, M. Linares y A. Milla, «Predicción de sigatoka negra en Plátano Hartón'bajo condiciones de fertilización del suelo en el municipio Veroes del estado Yaracuy, Venezuela,» *Agronomía Tropical*, vol. 59, nº 3, pp. 317-322, 2009.
- [34] E. Vargas, «La medición y vigilancias de la enfermedad de la mancha foliar producida por la sigatoka negra (*Mycosphaella fijiensis* var *diformis* Morelet) en la producción de muáceas: un enfoque para lograr la sostenibilidad del desarrollo,» *Revista Tecnología en Marcha*, vol. 21, nº 1, pp. 89-96.
- [35] G. Owomugisha, J. Quinn, E. Mwebaze y J. Lwasa, «Automated Vision-Based Diagnosis of Banana Bacterial Wilt Disease and Black Sigatoka Disease,» *Proceedings of the first International Conference on the use of Mobile ICT* , 2014.
- [36] B. Tigadi y B. Sharma, «Banana plant disease detection and grading using image processing,» *International Journal of Engineering Science and Computing*, pp. 6512-6516, 2016.
- [37] «Aprende Machine Learning,» 2017. [En línea]. Available: <https://www.aprendemachinelarning.com/7-pasos-machine-learning-construir-maquina/>. [Último acceso: 18 03 2019].
- [38] R. Gonzalez y R. Woods, *Procesamiento digital de imágenes*, 1996.
- [39] I. Fodor, «A survey of dimension reduction techniques,» 2002.
- [40] S. Russell y . P. Norvig, *Inteligencia Artificial: Un Enfoque Moderno*, 2009.
- [41] I. Goodfellow, Y. Bengio y A. Courville, *Deep Learning*, Massachusetts, 2016.
- [42] M. Frigo y S. Johnson, «FFTW: An adaptive software architecture for the FFT,» *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98*, pp. 1381-1384, 1998.
- [43] J. Terrillon, M. David y S. Akamatsu, «Automatic detection of human faces in natural scene images by use of a skin color model and of invariant moments,» *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 112-117, 1998.
- [44] R. Gonzalez y R. Woods, *Digital image processing*, Prentice hall New Jersey, 2002.

- [45] . B. Schölkopf y A. Smola, Learning with kernels: support vector machines, regularization, optimization, and beyond, MIT press, 2001.
- [46] R. Murray y S. Larry, Estadística, 2009.
- [47] A. Géron, Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow, 2019.
- [48] N. Altam, «An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression,» *The American Statistician*, vol. 46, nº 3, pp. 175-185, 1992.
- [49] M. Prabukumar, B. Krishna y J. Kamalakannan, «Image Processing and Pattern Classification Technique in a Machine Vision System that Identifies and Classifies the Plant Diseases Based on the Visual Symptoms,» *International Journal of Advanced Research in Computer Science*, vol. 1, nº 4, 2010.
- [50] Python, «PyWavelets,» [En línea]. Available: <https://pywavelets.readthedocs.io/en/latest/ref/dwt-discrete-wavelet-transform.html>. [Último acceso: 01 04 2019].
- [51] Numpy, «SciPy.org,» [En línea]. Available: <https://docs.scipy.org/doc/numpy/reference/generated/numpy.fft.fft.html>. [Último acceso: 01 04 2019].
- [52] Numpy, «SciPy.org,» [En línea]. Available: <https://docs.scipy.org/doc/numpy/reference/generated/numpy.histogram.html>. [Último acceso: 01 04 2019].
- [53] MathWorks. [En línea]. Available: <https://www.mathworks.com/help/stats/corr.html>. [Último acceso: 05 04 2019].
- [54] A. Bejarano, Calvo, Andrés y C. Henao, «Descriptores espacio-frecuencia para identificación automática de patrones de textura en productos textiles utilizando aprendizaje supervisado,» *Ciencia e Ingeniería Neogranadina*, vol. 28, nº 2, pp. 63-82, 2018.
- [55] K. Kenji y L. Rendell, «The feature selection problem: Traditional methods and a new algorithm,» *AAAI-92 Proceedings*, vol. 2, pp. 129-134, 1992.
- [56] S. Learn, «Ensemble methods,» [En línea]. Available: <https://scikit-learn.org/stable/modules/ensemble.html#id30>. [Último acceso: 03 04 2019].
- [57] A. Singh, B. Ganapathysubramanian, S. Sarkar y A. Singh, «Deep Learning for Plant Stress Phenotyping: Trends and Future Perspectives,» *Trends in plant science*, vol. 23, nº 10, pp. 833-898, 2018.
- [58] L. Rasdi, M. Fanany y A. Aniaty, «Metaheuristic Algorithms for Convolution Neural Network. Computational Intelligence and Neuroscience,» *Computational Intelligence and Neuroscience*, 2016.

- [59] B. Xu, N. Wang, C. Tianqi y M. Li, «Empirical Evaluation of Rectified Activations in Convolutional Network,» 2015.
- [60] W. Liu, Y. Wen, Z. Yu y M. Yang, «Large-margin softmax loss for convolutional neural networks,» *ICML*, vol. 2, nº 3, p. 7, 2016.