

**PROCESAMIENTO DE IMÁGENES MULTIESPECTRALES ADQUIRIDAS CON
DRONES PARA LA CARACTERIZACIÓN DE SISTEMAS AGROAMBIENTALES.**

**DAVID DANIEL HERNÁNDEZ MOLINA
JULIO MANUEL GULFO GALARAGA**

**UNIVERSIDAD PONTIFICIA BOLIVARIANA
ESCUELA DE INGENIERÍAS Y ARQUITECTURA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
MONTERÍA
2021**

**PROCESAMIENTO DE IMÁGENES MULTIESPECTRALES ADQUIRIDAS CON
DRONES PARA LA CARACTERIZACIÓN DE SISTEMAS AGROAMBIENTALES.**

**DAVID DANIEL HERNÁNDEZ MOLINA
JULIO MANUEL GULFO GALARAGA**

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
PROGRAMA DE INGENIERÍA ELECTRÓNICA
MONTERÍA
2021**

AGRADECIMIENTOS

Primeramente, damos gracias a Dios por darnos la oportunidad de cumplir satisfactoriamente los objetivos propuestos en este proyecto y poder culminar nuestro trabajo de grado.

A nuestra directora Claudia Milena Serpa Imbett y a los ingenieros del programa de ingeniería electrónica en la Universidad Pontificia Bolivariana seccional Montería por las asesorías y por su apoyo, durante todo nuestro pregrado.

A la Universidad Pontificia Bolivariana seccional Montería, por brindarnos una educación de calidad y por formarnos integralmente lo cual será de gran importancia durante toda nuestra vida profesional y personal.

A nuestros familiares y amigos por apoyarnos siempre en las buenas y en las malas.

Esta investigación fue apoyada con recursos propios de la Universidad Pontificia Bolivariana Sede Montería bajo el Proyecto Innova N ° 228-07 / 18-G007, del Proyecto Temático “DRIoT: Agricultura sostenible mediante el monitoreo en línea de las condiciones ambientales de cultivos y el uso racional de recursos hídricos y energéticos en sistemas de riego”.

LISTA DE CONTENIDO

RESUMEN	8
1. INTRODUCCIÓN	10
2. MARCO TEÓRICO/ESTADO DEL ARTE.	13
2.1 MARCO DE ANTECEDENTES.	13
2.2 MARCO TEÓRICO.	13
2.2.1 ESPECTRO ELECTROMAGNÉTICO.	13
2.2.2 TRANSFORMADA DE FOURIER.	14
2.2.3 FILTRADO DE IMÁGENES.	15
2.2.4 COMPRESIÓN.	15
2.2.5 SEGMENTACIÓN.	17
A. ALGORITMO DE UMBRALIZACIÓN.	17
B. ALGORITMO K-MEANS (K-MEDIAS)	19
C. ALGORITMO BASADO EN ECUACIÓN DE MAHALANOBIS.	20
D. TRANSFORMADA DE HOUGH	21
E. OPERADOR LAPLACIANO	21
F. GRADIENTE	21
G. MODELO LINEAL PARA LA ESTIMACIÓN DE LA PRODUCTIVIDAD.	22
2.3 MARCO CONCEPTUAL.	22
2.3.1 AGRICULTURA DE PRECISIÓN.	22
2.3.2 AGRICULTURA DE PRECISIÓN CON DRONES.	23
2.4 CÁMARA PARA DRONES.	23
2.4.1 CÁMARAS RGB	23
2.4.2 CÁMARA MULTIESPECTRAL.	23
2.4.3 CÁMARA HIPERESPECTRAL.	23
2.5 NDVI.	24
2.6 NDRE.	24
3. METODOLOGÍA.	26
3.1 DISEÑO METODOLÓGICO.	26
3.1.1 FASE 1: ALGORITMOS DE SEGMENTACIÓN BASADOS EN PROCESAMIENTO DE IMÁGENES.	26
3.1.2 FASE 2: SOFTWARE PROPIO: INTERFACES Y FUNCIONAMIENTO.	27
3.1.3 FASE 3: COMPROBACIÓN DE ALGORITMOS CON IMÁGENES SATELITALES.	32

3.1.4	FASE 4: NDVI Y SU RELACIÓN CON EL MODELO LINEAL DE PRODUCTIVIDAD, AHORRO Y GASTO.	33
4.	RESULTADOS Y DISCUSIÓN.	34
4.1	IMÁGENES SATELITALES UTILIZADAS PARA EL ANÁLISIS.	34
4.2	ESTIMACIÓN DE LA PRODUCTIVIDAD UTILIZANDO IMÁGENES LIGERAMENTE NO UNIFORMES.	39
4.3	ESTIMACIÓN DE LA PRODUCTIVIDAD MEDIANTE IMÁGENES NO MUY UNIFORMES.	42
5.	CONCLUSIONES Y RECOMENDACIONES.	45
6.	BIBLIOGRAFÍA	47
7.	ANEXOS.	50

LISTA DE FIGURAS

Figura 1. Espectro de una señal..	14
Figura 2. Filtrado de imágenes usando Matlab.	15
Figura 3. Compresión de imágenes usando Matlab.	16
Figura 4. Metodología propuesta.	26
Figura 5. Códigos base de procesamiento de imágenes en Matlab.	27
Figura 6. Interfaz para cargar imágenes a procesar.	28
Figura 7. Interfaz para la selección de zonas NDVI.	28
Figura 8. Interfaz para procesar contornos.	29
Figura 9. Interfaz para procesar color.	30
Figura 10. Interfaz para procesar imágenes usando el algoritmo de umbralización por áreas rectangulares.	30
Figura 11. Interfaz para procesar imágenes mediante el uso de umbrales por algoritmo de píxeles.	31
Figura 12. Interfaz para procesar imágenes mediante el uso de distancia de Mahalanobis.	32
Figura 13. Imagen NDVI procesada del software propio vs Imagen NDVI página oficial de Matlab.	32
Figura 14. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 50 m / píxel con alto contenido de rojo.	34
Figura 15. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 40 m / píxel con alto contenido de rojo.	35
Figura 16. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 40 m / píxel con alto contenido de amarillo.	35
Figura 17. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 40 m / píxel con alto contenido de amarillo.	36
Figura 18. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 40 m / píxel con alto contenido de verde.	36
Figura 19. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 40 m / píxel con alto contenido de verde.	37
Figura 20. Imágenes NDVI para probar el modelo lineal de productividad de rendimiento en imágenes altamente no uniformes con resolución de 50 m / píxel.	37
Figura 21. Imágenes NDVI para probar el modelo lineal de productividad de rendimiento en imágenes altamente no uniformes con resolución de 40 m / píxel.	38
Figura 22. Imágenes NDVI para probar el modelo lineal de productividad de rendimiento en imágenes altamente no uniformes con resolución de 40 m / píxel.	38

LISTA DE TABLAS

Tabla 1. Productividad asignada (en porcentaje) según modelo lineal.	33
Tabla 2. Ahorro asignado (en porcentaje) según modelo lineal.	33
Tabla 3. Gasto asignado (en porcentaje) según modelo lineal.	33
Tabla 4. Valores de productividad estimados mediante modelo lineal, segmentación por procesamiento de imágenes en imágenes de la Figura 14-19.	40
Tabla 5. Valores de ahorro estimados mediante modelo lineal, segmentación por procesamiento de imágenes de la Figura 14-19	40
Tabla 6. Valores de gasto estimados mediante modelo lineal, segmentación por procesamiento de imágenes de la Figura 14-19.	41
Tabla 7. Valores de productividad estimados mediante el modelo lineal, segmentación por procesamiento de imágenes en imágenes de la figura 20-22.	43
Tabla 8. Valores de ahorro estimados mediante el modelo lineal, segmentación por procesamiento de imágenes en imágenes de la figura 20-22.	43
Tabla 9. Valores de gasto estimados mediante el modelo lineal, segmentación por procesamiento de imágenes en imágenes de la figura 20-22.	44

RESUMEN

En este proyecto se implementó un sistema de procesamiento de imágenes multiespectrales en Matlab AppDesigner 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; específicamente, el método de umbralización, K-means y Mahalanobis,.

A partir de la aplicación de estos algoritmos, se aplicó una metodología para estimar el comportamiento porcentual de variables agrícolas relacionadas con la productividad, ahorro o gasto, las cuales están directamente relacionada con la variabilidad del NDVI en la imagen procesada. Se encontró que si la imagen es ligeramente uniforme todos los métodos de procesamiento dan resultados muy similares, mientras que en el caso contrario; es decir, cuando la imagen no es uniforme hay desviaciones en los resultados entre los diferentes algoritmos, por lo que se propone un criterio para la escogencia del método de cálculo para estimación de las variables agrícolas.

Palabras Claves: Agricultura de precisión, Procesamiento de Imágenes, Segmentación de Imágenes, NDVI, Matlab.

ABSTRACT

In this project, a multispectral image processing system was implemented in Matlab AppDesigner from multispectral cameras or satellite images that calculate the value of NDVI (Normalized Vegetation Index Difference) 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; specifically, the thresholding method, K-means and Mahalanobis.

From the application of these algorithms, a methodology was applied to estimate the percentage behavior of agricultural variables related to productivity, savings or expenditure, which are directly related to the variability of the NDVI in the processed image. It was found that if the image is slightly uniform, all the processing methods give very similar results, while in the opposite case; that is, when the image is not uniform, there are deviations in the results between the different algorithms, for which a criterion is proposed for the choice of the calculation method for estimating the agricultural variables.

Keywords: precision agriculture, Image processing, Image segmentation, NDVI, Matlab.

1. INTRODUCCIÓN

La analítica de Datos tiene un gran potencial para impulsar la Agricultura de Precisión (Dongdong, y otros, Stress Distribution Analysis on Hyperspectral Corn Leaf Images for Improved Phenotyping Quality, 2020). Estos datos podrían ser recolectados usando diferentes tecnologías tales como, imágenes, sensores, estaciones meteorológicas, datos geográficos, etc. Este conjunto de tecnologías aplicadas al campo reúne la información necesaria para entender las variaciones del suelo, y estimar la salud de los cultivos.

Uno de los métodos recientemente usados en Agricultura de Precisión es el análisis de imágenes obtenidas con cámaras especializadas que tienen la capacidad de proveer información acerca de la salud de los cultivos, o caracterizar la aridez de terrenos usados en agricultura. Algunas de estas cámaras especializadas se fabrican con arreglos de fotodetectores que capturan la luz reflejada a cierta longitud de onda en el espectro visible hasta el infrarrojo. Particularmente, si capturan individualmente más de una longitud de onda se les llama cámaras multiespectrales, y si capturan un continuo de longitudes de onda se les conoce como cámaras hiperespectrales. Según la cámara usada, así será clasificada la tipología de la imagen; así que, si la imagen se captura con una cámara convencional de fotografía, esta se considera como RGB (Red-Green-Blue, por sus siglas en ingles), alternativamente si son usadas las cámaras especializadas mencionadas, las imágenes se conocen como multiespectrales o hiperespectrales. A partir del procesamiento de dichas imágenes, en términos de los valores de intensidad de radiación detectada por el arreglo de fotodetectores de la cámara, se calculan unos índices normalizados, por ejemplo, para la caracterización de la vegetación o terrenos. Uno de ellos basadas en un cálculo normalizado de la luz detectada, por ejemplo, con cámaras multiespectrales diseñadas con al menos un detector en la longitud de onda del rojo (633 nm) y otro en el infrarrojo (850 nm), se conoce como imagen NDVI (Normalized Difference Vegetation Index, por sus siglas en ingles), y son usadas para inferir el estado de la salud de la planta (Candiago , Remondino , de Giglio, Dubbini, & Gattelli, 2015) (Haggar, Stent, & Rose, 1984); dichas imágenes pueden ser adquiridas ya sea por satélite, vehículos aéreos no tripulados (VANT) o drones (Bhattarai, Bajgiran, Weiskittel, & MacLean, 2020) (Moreira, y otros). Estas imágenes, apoyan los procesos de Agricultura de Precisión definida como: *"una estrategia de manejo que recolecta, procesa y analiza datos temporales, espaciales e individuales y los combina con otra información para apoyar las decisiones de manejo de acuerdo con la variabilidad estimada, y así mejorar la eficiencia en el uso de los recursos, la productividad, la calidad, la rentabilidad y la sostenibilidad de la producción agrícola"*. El análisis de datos de esta tecnología multiespectral puede ser una base para el desarrollo de modelos predictivos de costos y riesgos asociados a proyectos de Agricultura de Precisión (Agricultura 4.0), de acuerdo con las necesidades particulares de los agricultores. Es por esto por lo que, en este trabajo se procesaron imágenes NDVI, con el fin de proponer un modelo para el conocimiento de variables agrícolas en un terreno según la variabilidad espacial del NDVI, que permita en un futuro, ser la base de sistemas de gestión de cultivos. Se propuso que a partir de una

adecuada correlación entre los datos NDVI procesados, apoyados de la información de los requerimientos de los agricultores, se podría contribuir al éxito en la implementación de sistemas de Agricultura de Precisión basada en procesamiento de imágenes NDVI, y así, el propósito es que los agricultores, y otros actores aseguren una coproducción efectiva (Li, y otros, 2020). En cuanto a las decisiones de manejo a partir del análisis y monitoreo de imágenes procesadas, recientemente se han abordado métodos para sistematizar y mejorar la eficiencia en el uso de recursos basados en imágenes NDVI, los cuales buscan aumentar la productividad de la producción agrícola (Mukesh Singh, Komal, & Victorovich, 2020) (Dongdong, y otros, Precise Estimation of NDVI with a Simple NIR Sensitive RGB Camera and Machine Learning Methods for Corn Plants, 2020) (Choudhary, Shi, Boori, & Corgne, 2019) (Hasseb, Din, Almogren, & Islam, 2020). Sin embargo, en una revisión del estado del arte de esta tecnología, se observó que existen pocas investigaciones que utilicen imágenes NDVI para correlacionar la información de la imagen con las variables prácticas de interés de utilidad en Agricultura de Precisión (dos Santos, Madalozzo, Cunha Fernandes, & Rieder, 2020) (Ji, Bai, Qiao, Wang, & Chang, 2020). Es notoria las potencialidades del procesamiento de imágenes en Agricultura de Precisión, pero el impulso de dichas potencialidades dependen no solo del análisis de la imagen misma, sino de la forma de correlacionar estas imágenes con las variables agrícolas de interés para quienes desempeñan un rol de sistematización, análisis y toma de decisión. Este aspecto es uno de los propósitos de este trabajo de grado. En este documento, se presentarán los resultados del desarrollo de un software para Agricultura de Precisión, registrado ante el DNDA (Dirección Nacional de Derechos de Autor) a la fecha de presentación de este informe, el cual fue llamado como DRIoT: Software para Agricultura de Precisión. En este, se analizó el desempeño de diferentes algoritmos de procesamiento de imágenes aplicados a imágenes NDVI obtenidas a partir de satélites. Estos algoritmos se basan en códigos de segmentación desarrollados e integrados en un software propio en Matlab AppDesigner, y se basan en la detección de color en la imagen, utilizando tres técnicas: umbralización con detección de zonas rectangulares, umbral por detección de píxeles y distancia de Mahalanobis. Finalmente, se correlacionó el contenido NDVI de la imagen con variables agrícolas como el ahorro de agroquímicos/agua, gasto de recursos también, por ejemplo, agroquímico/agua y productividad. Basadas en el resultado de los algoritmos de segmentación de imágenes, se planteó un modelo lineal en función de la distribución de valores NDVI y su correlación para la estimación de variables agrícolas. Se compararon los resultados en términos de desviación porcentual de los modelos de umbralización con detección de zonas rectangulares y distancia de Mahalanobis, respecto al resultado de referencia: umbral por detección de píxeles. Se mostró que dichos resultados de estimación de variables agrícolas son más precisos en imágenes con alto grado de uniformidad comparada con imágenes no uniforme. Se espera que dicha herramienta pueda ser implementada junto con otros datos analíticos, por ejemplo, como los provenientes de sensores para Agricultura de Precisión (estaciones meteorológicas humedad, temperatura, etc.) para incentivar el uso de estos métodos por los actores del sector agrícola, actuantes en diferentes roles, interesados por ejemplo en los planes de manejo de agroquímicos, recursos

energéticos y ahorro de agua, que se interesen en el análisis imágenes multiespectrales.

2. MARCO TEÓRICO/ESTADO DEL ARTE.

En esta sección se encontrarán las definiciones de las diferentes componentes teóricas que se usaron para llevar a cabo este proyecto. La sección 2.1 se encuentran algunos antecedentes que se encontraron 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 trabajo de grado: “Aplicaciones de análisis de imágenes en agricultura de precisión” (Moreno, Tangarife , & Díaz , Aplicaciones de análisis de imágenes en agricultura de precisión, 2017). En este, se puede encontrar un trabajo similar al que se presenta en este informe ya que se realizó una aplicación de procesamiento de imágenes en el que se analizaban diferentes factores como preparación del suelo, límites y áreas de tierra, monitoreo de vegetación, clasificación de la vegetación, crecimiento, altura, estado fitosanitario; manejo de enfermedades, plagas y malezas, para generar un método de Agricultura de Precisión.

Se encontró un trabajo de investigación el cual tiene como título “Procesamiento de imágenes multiespectrales en cultivos de café y cacao” (Gonzáles, Gómez, & Amaya, 2017). En este trabajo se puede encontrar el procesamiento de imágenes aplicado en los cultivos de café y cacao en la región de Santander, donde usan el software de programación Matlab para brindar al usuario una interfaz gráfica (GUI) en las cuales se pueden observar las distintas mediciones que se realizaron como: Umbral de vegetación, El NDVI, El estrés hídrico y el CIR.

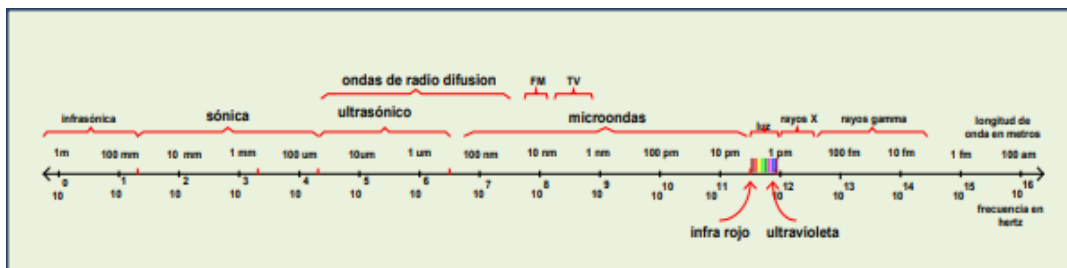
2.2 MARCO TEÓRICO.

2.2.1 Espectro electromagnético.

El espectro es una representación en el dominio de la frecuencia de las ondas electromagnéticas. El eje horizontal representa la frecuencia con una variación logarítmica que va desde 1 Hz hasta los 10^{20} Hz.

En la Figura 1 se observa que el eje de frecuencia está rotulado por las aplicaciones más frecuentes en el área de electrónica, y de izquierda a derecha: frecuencias infrasónicas, frecuencias audibles o sónicas, frecuencias ultrasónicas, frecuencias que corresponden a la transmisión de las señales de audio y de video, frecuencias que corresponden a las microondas, frecuencias ópticas, y a las frecuencias superiores a las de la luz como son los rayos x y rayos gamma. (Arellano, 2014).

Figura 1. Arellano, R. (2014, diciembre 15). Espectro de una señal. Recuperado de <http://asdopen.unmsm.edu.pe/files/Revista6.pdf#page=5>.



2.2.2 Transformada de Fourier.

La transformada de Fourier, denominada así por su autor: Joseph Fourier, es una transformación matemática en la cual interviene una base ortogonal de funciones periódica, que permite proyectar cada componente frecuencial de la señal en dicha base, a partir de la cual se obtiene una función o señal en “otro dominio”. En nuestra aplicación, un transformada de Fourier convierte una señal en el dominio del tiempo, al dominio de la frecuencia. En la transformada de Fourier se obtiene para cualquier función periódica o no, una función que permite analizar la respuesta en frecuencia de una señal o sistema. Matemáticamente, la ecuación (1) modela la transformada de Fourier (Soliman & Srinath , 1999):

$$F(w) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt \quad (1)$$

Donde:

- $f(t)$ es la función a la cual se le desea aplicar la transformada.
- $i = \sqrt{-1}$.
- $e^{-i\omega t} = \cos(\omega t) * (-i)\sin(\omega t)$.

Algunas de sus aplicaciones en la ingeniería son:

- Analizar el contenido de frecuencias de las señales.
- Determinar cómo cambia la amplitud y las fases de señales sinusoidales cuando estas pasan a través de un sistema lineal e invariante en el tiempo.
- Estudiar la respuesta en el tiempo de una variante de un circuito eléctrica donde la señal de entrada no es sinusoidal ni cosenoidal.

En el campo que más se usa es:

- Comunicaciones.
- Ingeniería Mecánica.
- Ingeniería de control.
- Procesamiento de señales de audio.
- Procesamiento de imágenes.

- En el área médica. (Adam, Amaya, & Melendez , 2012)

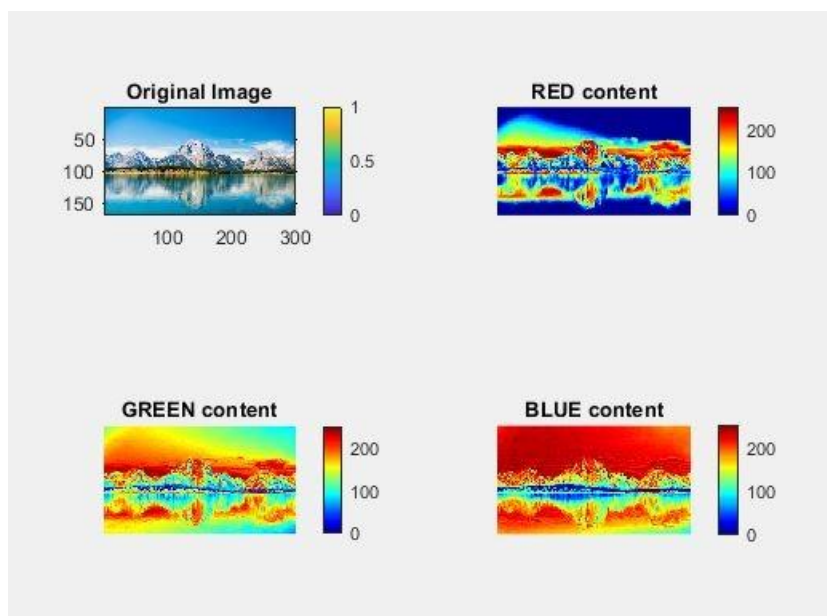
2.2.3 Filtrado de imágenes.

Es el proceso por el que la parte esencial o útil de una señal se separa de otras componentes extrañas o indeseadas que se denominan generalmente ruido. El termino ruido que se usa se refiere a cualquier parte no deseada de una señal (Sánchez, 2009).

Un filtro ideal selectivo en frecuencia es aquel que deja pasar ciertas frecuencias inalteradas y elimina el resto. En este proceso de filtrado se utiliza la Transformada de Fourier que permite obtener la respuesta de la señal temporal en el dominio de la frecuencia, con el fin de realizar filtrado a partir de multiplicaciones con los filtros o convoluciones. El intervalo de frecuencias que pasan se denomina banda de paso del filtro y el intervalo de frecuencias que se elimina se denomina banda eliminada (Soliman & Srinath , 1999).

Los filtros se utilizan para realizar operaciones con el fin de obtener un resultado deseado, como por ejemplo, en la Figura 2 se puede visualizar distintos tipos de filtros que se pueden aplicar en una misma imagen, conteniendo así un parametro de la imagen original el cual se desea eliminar para un posterior análisis.

Figura 2. Autoría propia [2020]. Filtrado de imágenes usando Matlab.



2.2.4 Compresión.

La compresión de imagen digital juega un papel extremadamente importante en la transmisión y almacenamiento de datos de imagen digital. Generalmente La cantidad de datos asociados con la información visual es tan grande que su

almacenamiento requiere una memoria enorme y su transmisión requiere altos anchos de banda. La compresión de imagen es el proceso de codificación efectiva de imágenes digitales para reducir el número de bits necesarios para representar una imagen. Esta compresión permite la transmisión de imágenes a anchos de banda muy bajos y minimiza los requisitos de espacio para el almacenamiento de estos datos.

Como fotogramas consecutivos de imágenes fijas constituyen el video. Los datos y algoritmos diseñados para la compresión de imágenes fijas 2D, pueden ser extendidos efectivamente para comprimir datos de video. Los algoritmos de compresión pueden clasificarse ampliamente en 'sin pérdida' y algoritmos de compresión "con pérdida". Una compresión sin pérdidas reproduce la imagen original exactamente sin ninguna pérdida de información, como se puede observar en la figura 3. Estos métodos se utilizan en aplicaciones donde la pérdida de datos es inaceptable, por ejemplo, datos de texto e imágenes médicas. Mientras que la compresión 'con pérdidas' es donde la imagen reproducida no es exacta, una réplica de la imagen original. Alguna información se pierde en el proceso de codificación. Estos algoritmos proporcionan una forma de compensación entre la calidad de la imagen y el grado de compresión. El objetivo es lograr un mayor grado de compresión sin mucha degradación en la calidad de la imagen (Hussain & Jeong, 2015).

En la figura 3 se muestra el resultado final de una compresión realizada a una imagen seleccionada anteriormente. En este proceso la imagen se divide en distintas partes por medio de filtros, luego se toman esas partes y se concatenan por medio de la compresión, logrando así una fidelización con un grado sobresaliente, con respecto a la imagen original.

Figura 3. Autoría propia [2020]. Compresión de imágenes usando Matlab.



2.2.5 Segmentación.

El proceso de partición de una imagen digital en varias regiones (conjunto de píxeles) se denomina segmentación de imagen. Dichas particiones se realizan estableciendo unos criterios que pueden depender del color, textura, entre otros. Las particiones se consideran zonas diferentes en la imagen que tienen la misma textura o color. El resultado de la segmentación de la imagen es un conjunto de regiones (similar a una rompecabeza), que representan colectivamente toda la imagen. Todos los píxeles de una región son similares con respecto a alguna característica o propiedad predeterminada, como el color, la intensidad o la textura. Las regiones adyacentes son significativamente diferentes con respecto a las mismas características. Algunas de las aplicaciones prácticas de la segmentación de imágenes son: procesamiento de imágenes, visión artificial, reconocimiento facial, imágenes médicas, bibliotecas digitales, recuperación de imágenes y videos (Puranik, P, & Palsodkar, 2009). Existen varios métodos de segmentación y se mostraran los fundamentos teóricos usados en la implementación del software, los cuales se basan en algoritmos de umbralización, k-means y Mahalanobis para la segmentación de imágenes mientras que para la detección de contornos se basó en algoritmos de Hough, Laplace y Gradiente.

A. Algoritmo de umbralización.

La umbralización es una técnica de segmentación simple y eficiente que permite separar los píxeles de una imagen según un valor umbral en dos categorías a partir de un valor de intensidad del píxel (Triana, Jaramillo, Gutiérrez, & Rodríguez, 2016). La umbralización se puede definir mediante la función matemática (vease ecuación 3)

$$T = T[x, y, p(x, y), f(x, y)] \quad (3)$$

donde:

- x e y son las coordenadas del píxel en la imagen f .
- $f(x, y)$ es la intensidad del píxel en dichas coordenadas.
- $p(x, y)$ representa alguna propiedad local del punto que vamos a tener en cuenta para la discriminación.

A partir de estos parámetros, se define una función $g(x, y)$ (vease ecuación 4).

$$g(x, y) = \begin{cases} 0 & \text{si } f(x, y) > T \\ 1 & \text{si } f(x, y) \leq T \end{cases} \quad (4)$$

Donde:

- en $g(x,y)$ se observa que los píxeles a los que se les asigna un valor 0 corresponden a los objetos, mientras que los que corresponden al entorno tienen valor 1.

En este proyecto se usaron dos tipos de umbralización: por áreas y por pixel implementados en Matlab®. los cuales cuentan con las líneas de código descritas más adelante, y que serán el “core” de su principal funcionamiento.

Con el código de umbralización por áreas lo que se busca es transformar la imagen base a trabajar en blanco y negro para así poder aplicarle el valor umbral deseado siguiendo esto se genera una matriz vacía en donde los datos que están dentro del valor umbral que encuentra se almacena dentro de tal matriz, para luego obtener un promedio y así calcular la parte de la imagen con un umbral parecido.

Parte del código implementado se muestra (sic) aquí:

```
I2=imbinarize(imagen5,'adaptive','Sensitivity',0.5);
Iumbral=bwlabel(I2);
a=max(max(Iumbral));
a2=0;
promedios1=[];
maxmin=[];
for etiqueta=1:a
    [k,l]=find(Iumbral==etiqueta);
    m=1; n=0;
    xmin=min(k); ymin=min(l);
    xmax=max(k); ymax=max(l);
    x=xmax-xmin+1; y=ymax-ymin+1;
    Irecup=imagen5(xmin:xmax,ymin:ymax);%I
    if(size(Irecup)>30)
        a2=a2+1;
        Isegment1{a2}=Irecup;
        prom=mean(mean(Irecup));
        promedios1=[promedios1 prom];
        maxp=max(max(Irecup));
        minp=min(min(Irecup));
        maxmin(1,a2)=maxp;
        maxmin(2,a2)=minp;
```

Para la umbralización por pixel también se transforma la imagen a blanco y negro, luego se definen los valores de umbrales deseados para así encontrar dos resultados diferentes para hacer la intersección de estos y generar una nueva matriz que sería la resultante.

Parte del código implementado se muestra (sic) aquí:

```
limiteinf=app.Limiteinf.Value;
limitesup=app.Limitesup.Value;
```

```

I4=imbinarize(I,limitesup);
I5=imbinarize(I,limiteinf);
I6=not(I4);
I7=and(I6,I5);

```

B. Algoritmo k-means (k-medias)

k-means o k-medias, en español, es uno de los algoritmos más utilizados para realizar agrupamiento. Dicha técnica es implementada en Minería de Datos. La idea del k-medias es colocar todos los objetos en un espacio determinado y dadas sus características formar grupos de objetos con rasgos similares pero diferentes a los demás que integran otros grupos (Lara Gutiérrez , Lara Ruiz, Hernández Hernández, Hernández Hernández, & Hernández Hernández , 2016).

Este algoritmo se puede modelar matemáticamente mediante algunos valores relacionados por la ecuación (véase ecuación 5).

$$J = \sum_{i=1}^m \sum_{k=1}^K w_{ik} \left| |x^i - \mu_k| \right|^2 \quad (5)$$

Donde:

- μ_k es la media de puntos.
- x^i es el conjunto de datos.
- k es el subconjunto de puntos entre el conjunto de datos originales.

De manera computacional el código de segmentación usando la técnica de k-means es el siguiente:

Como primer paso, se lee la imagen RGB luego se inicializa el almacenamiento para cada región de muestra, se selecciona cada region de ejemplo y se convierte la imagen RGB a una imagen CIE L*a*b* que es un espacio de color uniforme usado para evaluar el color de un objeto, donde L* indica la luminosidad, a* las coordenadas rojo/verde y b* las coordenadas amarillo/azul, para así calcular el valor medio de a* y b* para cada area y clasificar cada pixel.

Parte del código implementado se muestra (sic) aquí:

```

for count = 1:nClasses
set(f, 'name', ['select sample region for ' classes{count}]);
sample_regions(:, :, count) = roipoly(x);
end
cform = makecform('srgb2lab');
lab_x = applycform(x, cform);

color_labels = 0:nClasses-1;
a = double(a);
b = double(b);

```

```
distance = repmat(0,[size(a),nClasses]);

purple = [119/255 73/255 152/255];
plot_labels = {'k', 'r', 'g', purple, 'm', 'y'};
```

C. Algoritmo basado en ecuación de Mahalanobis.

Está basada en correlaciones entre variables con diferentes patrones que pueden ser identificados y analizados. Es una manera útil de determinar la similitud de un conjunto de muestras desconocida con respecto a uno conocido. Difiere de la distancia Euclídea en que tiene en cuenta las correlaciones entre el conjunto de datos (vease ecuación 6) (Cárdenes & Ruiz Alzola, 2005).

$$D = \sqrt{(X - Y)^T * \Sigma^{-1}(X - Y)} \quad (6)$$

Donde:

- Σ^{-1} a matriz de covarianzas no singular.
- $(x - y)^T$ es la transpuesta de los datos.

De manera computacional el código de segmentación usando la distancia de Mahalanobis se describe como:

Como primer paso, se debe seleccionar las coordenadas de la imagen donde se quiere realizar la segmentación, tomando así los tres componentes de colores RGB y luego se aplica la fórmula matemática creando así una nueva matriz.

Parte del código implementado se muestra (sic) aquí:

```
x = app.coordenadaX.Value;
y = app.coordenadaY.Value;
colormedio1=[IM(y,x,1); IM(y,x,2); IM(y,x,3)];
MC=[1 0 0;0 1 0;0 0 1];
MCi=inv(MC);
IM2=IM;
for f=1:nf
    for c=1:nc
        z=[IM(f,c,1); IM(f,c,2); IM(f,c,3)];
        a=colormedio1;
        z=double(z)/255;
        a=double(a)/255;
        d2=(z-a)'*MCi*(z-a);
        if d2>0.08
            IM2(f,c,:)=[0 0 0];
        end
    end
end
```

D. Transformada de Hough

Es una herramienta computacional que aísla características las cuales se utiliza para la detección de contornos en una imagen u objeto, su principal campo de estudio es para parametrizar líneas rectas, polinomios o círculos. (Urrea & Ospina, 2004)

Como primer paso, se debe convertir la imagen a blanco y negro, y con los comandos Edge y Hough detectan todos los bordes presentes en la imagen.

Parte del código implementado se muestra (sic) aquí:

```
I1=rgb2gray(I1)
Ih = edge(I1, 'approxcanny');
[H,T,R] = hough(BW, 'RhoResolution',0.5, 'ThetaResolution',0.5)
```

E. Operador Laplaciano

Es un operador matemático relacionado con la segunda derivada de una función multivariable. (Urrea & Ospina, 2004)

Se implementó así: Como primer paso, se debe convertir la imagen a blanco y negro, y con unos parámetros ya establecidos se pasan por unos filtros los cuales sirven para determinar los bordes de la imagen.

Parte del código implementado se muestra (sic) aquí:

```
I2 = rgb2gray(I);
G=[0 1 2 1 0;1 3 5 3 1;2 5 9 5 2;1 3 5 3 1;0 1 2 1 0];
G=G*(1/57);
IF1 = imfilter(I2,G);
L = [0 0 -1 0 0;0 -1 -2 -1 0;-1 -2 16 -2 -1;0 -1 -2 -1 0;0 0 -1 0 0]/16;
IF = imfilter(IF1,L)
```

F. Gradiente

Es un operador matemático relacionado con la primera derivada de una función multivariable.

Parte del código implementado se muestra (sic) aquí:

```
I2 = im2double(imagen);
I2 = rgb2gray(I2);
G=[0 1 2 1 0;1 3 5 3 1;2 5 9 5 2;1 3 5 3 1;0 1 2 1 0];
G=G*(1/57);
I2 = imfilter(I2,G);
[Dx,Dy] = gradient(I2);
IG=sqrt(Dx.^2+Dy.^2);
```

G. Modelo lineal para la estimación de la productividad.

Ahorro (en porcentaje) según modelo lineal

$$\%A_t = \frac{NDVI(i,j)_{(0-0.2)}(0) + NDVI(i,j)_{(0.21-0.4)}(0) + NDVI(i,j)_{(0.41-0.6)}(0) + NDVI(i,j)_{(0.61-0.8)}(40) + NDVI(i,j)_{(0.81-1)}(60)}{NDVI(i,:) * NDVI(:,j)} \quad (7)$$

Gasto (en porcentaje) según modelo lineal.

$$\%G_t = \frac{NDVI(i,j)_{(0-0.2)}(80) + NDVI(i,j)_{(0.21-0.4)}(60) + NDVI(i,j)_{(0.41-0.6)}(40) + NDVI(i,j)_{(0.61-0.8)}(0) + NDVI(i,j)_{(0.81-1)}(0)}{NDVI(i,:) * NDVI(:,j)} \quad (8)$$

Productividad (en porcentaje) según modelo lineal.

$$\%P_t = \frac{NDVI(i,j)_{(0-0.2)}(20) + NDVI(i,j)_{(0.21-0.4)}(40) + NDVI(i,j)_{(0.41-0.6)}(60) + NDVI(i,j)_{(0.61-0.8)}(80) + NDVI(i,j)_{(0.81-1)}(100)}{NDVI(i,:) * NDVI(:,j)} \quad (9)$$

Donde:

- $\%A_t$ es el ahorro total de la imagen.
- $\%G_t$ es el gasto total de la imagen.
- $\%P_t$ es la productividad total de la imagen.
- $NDVI(i,:) * NDVI(:,j)$ es el área de la imagen (número total de píxeles)
- $NDVI(i,j)$ matriz de la imagen NDVI

2.3 MARCO CONCEPTUAL.

2.3.1 Agricultura de precisión.

El objetivo de la Agricultura de precisión consiste en aplicar la cantidad correcta de insumos, en el momento adecuado y en el lugar exacto. Es el uso de instrumentos especializados para tecnologías de la información y comunicaciones, sensores, satélites, radares, drones, entre otros; son usados en la Agricultura de Precisión para estudiar el estado de un cultivo o terreno, y así adecuar el manejo de suelos y cultivos según la variabilidad de las características determinadas por dichos instrumentos. La Agricultura de Precisión involucra el uso de sistemas de posicionamiento global (GPS) y de otros medios electrónicos para obtener datos del cultivo. Las tecnologías de la agricultura de precisión permiten satisfacer una de las exigencias de la agricultura moderna: el manejo óptimo de grandes extensiones (García & Flego), así como la optimización de la producción de pequeños cultivos relacionados con granjas y cultivos comunitarios.

2.3.2 Agricultura de precisión con drones.

En los últimos años el uso de vehículos aéreos no tripulados ha sido de gran ayuda, ya que se utilizan en diferentes aspectos de la vida cotidiana, en las cuales se encuentran la silvicultura, geología, sector ganadero y seguridad, dentro de las aplicaciones más comunes se encuentran: en la industria del cine, riego de cultivos y adquisición de imágenes, entre otras, pero esta última aplicación es una de las más importantes, ya que permite el procesamiento de imágenes para así brindar una agricultura mucho más tecnificada que la convencional (Moreno, Tangarife, & Díaz, 2017).

2.4 CÁMARA PARA DRONES.

2.4.1 Cámaras RGB

Son cámaras que permiten visualizar las imágenes reales en términos de los colores primarios (Red (R)-Green (G)-Blue (B)), Rojo, Verde y Azul, por sus siglas en ingles), se pueden obtener numerosos colores 'X', por mezcla o suma aritmética. La naturaleza del color obtenido es trivariante, al ser los tres colores primarios independientes entre sí, es decir, un color 'X' de luminancia 'L', se puede igualar mediante la triada (R, G, B), o cantidades adecuadamente dosificadas de tres componentes primarias (Escalera, 2016).

2.4.2 Cámara Multiespectral.

Es un dispositivo que capta múltiples longitudes de onda, referida a cualquier componente del espectro en el dominio de la frecuencia. Esta cámara es diferente a la RBG, debido a que incluye arreglos de fotodetectores sensibles a la longitud de onda particular en la que opera. Permiten captar la intensidad de radiación de la longitud de onda particular en la que opera la cámara multiespectral; por ejemplo, longitudes de onda de tipo Red Edge (frontera del rojo) que va desde 680 a 750 nm, y el NIR (infrarrojo cercano) que va desde 750 nm a 1700 nm. Estas cámaras al ser usadas en Agricultura de Precisión permiten detecciones de propiedades para la caracterización de la salud de las plantas. La intensidad de radiación en la longitud de onda Red Edge e IR cercano está relacionada propiedades de absorción de las plantas relacionadas con la cantidad de clorofila, y por lo tanto su salud. Dependiendo del espectro o longitudes de onda, se caracterizan, por ejemplo: la fotosíntesis (490nm, 676 nm), celulosa (1220nm y 1930nm), contenido de agua (970nm, 1194nm y 1458 nm) (Peña, Rentería , Velásquez , Ojeada , & Barrera , 2019) (Insights, s.f.).

2.4.3 Cámara Hiperespectral.

Los sistemas de imágenes hiperespectrales se basan en espectrómetros que realizan un registro espectral, en un intervalo continuo de longitudes de onda. A partir de esta información se infiere acerca de la información química o física de

manera precisa y confiable. Las cámaras hiperespectrales están equipadas con sensores capaces de percibir centenares de longitudes de onda dentro y fuera del espectro visible, haciendo posible la generación de imágenes que contienen múltiples informaciones, en nuestro caso de los cultivos. Estas cámaras resuelven una amplia gama de tareas analíticas en procesos como clasificación, análisis de ingredientes o distribución (mesurex, s.f.).

2.5 NDVI.

El Índice de Vegetación de Diferencia Normalizada, también conocido como NDVI (Normalized Difference Vegetation Index, por sus siglas en inglés), es un índice de vegetación que se utiliza para estimar la cantidad, calidad y desarrollo de la vegetación con base a la medición de la intensidad de la radiación de ciertas bandas del espectro electromagnético que la vegetación emite o refleja.

Para el cálculo de los índices de vegetación se debe conocer la intensidad de la radiación reflejada para las longitudes de onda del rojo e infrarrojo cercana (660nm, 790nm respectivamente) (véase ecuación 7), de ese espectro electromagnético, y su ecuación para el cálculo se puede obtener a partir de la ecuación (1) (Alonso, s.f.)

$$NDVI = \frac{R_{nir} - R_{red}}{R_{nir} + R_{red}} \quad (10)$$

Donde:

- R_{nir} es la reflectancia del Infrarrojo Cercano.
- R_{red} es la reflectancia del rojo.

2.6 NDRE.

El Índice de borde rojo de diferencia normalizada, también conocido como NDRE (Normalized Difference Red Edge, por sus siglas en inglés) se calcula usando el espectro del Infrarrojo Cercano (NIR) y la reflectancia del borde rojo (Red-Edge) (véase ecuación 8). Este ha demostrado ser más ventajoso que el NDVI para optimizar los tiempos de cosecha basados en las transiciones de la actividad de la fotosíntesis (Maccioni, Agati, & Mazzinghi, 2001).

La razón es que la luz Red Edge (la que se usa en el NDRE) puede atravesar las hojas mucho más profundamente que la luz roja (la que se usa en el NDVI) (DIFERENCIA NORMALIZADA DE BORDE ROJO, s.f.). La ecuación (8) modela dicho valor:

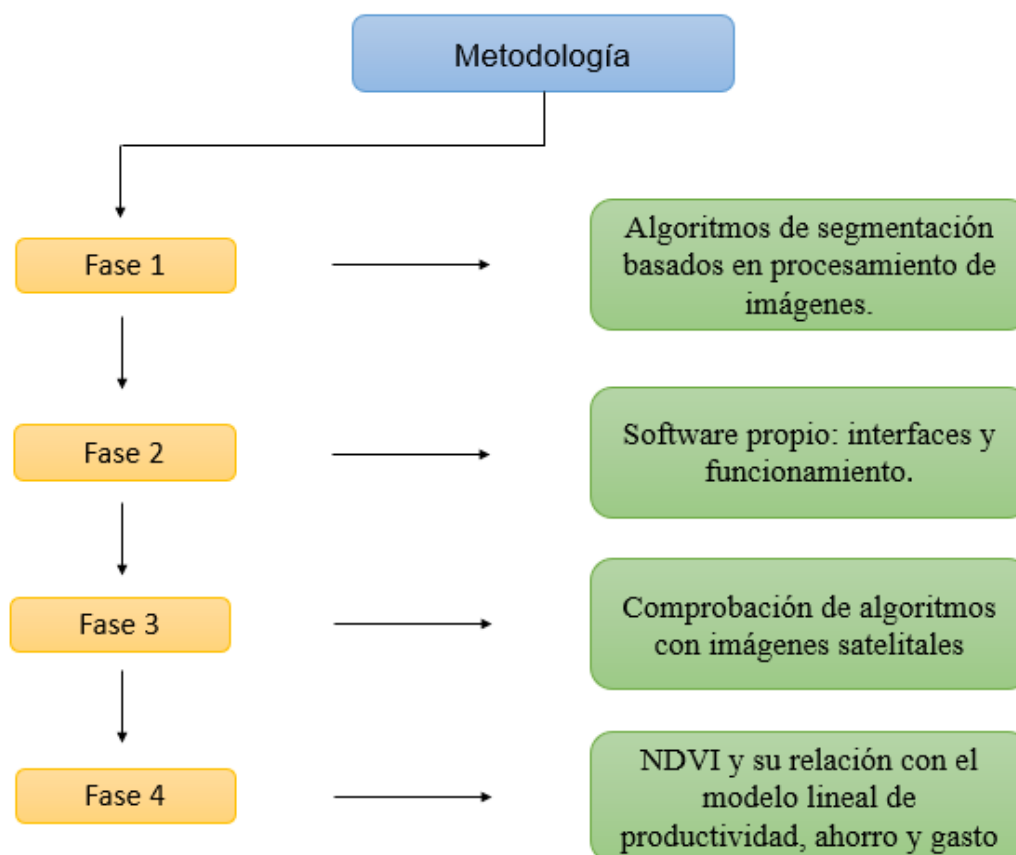
$$NDRE = \frac{R_{nir} - R_{red-edge}}{R_{nir} + R_{red-edge}} \quad (11)$$

Donde:

- R_{nir} es la reflectancia del Infrarrojo Cercano.
- $R_{red-edge}$ es la reflectancia en el borde rojo.

3. METODOLOGÍA.

Figura 4 Autoría propia [2020]. Metodología propuesta.



3.1 DISEÑO METODOLÓGICO.

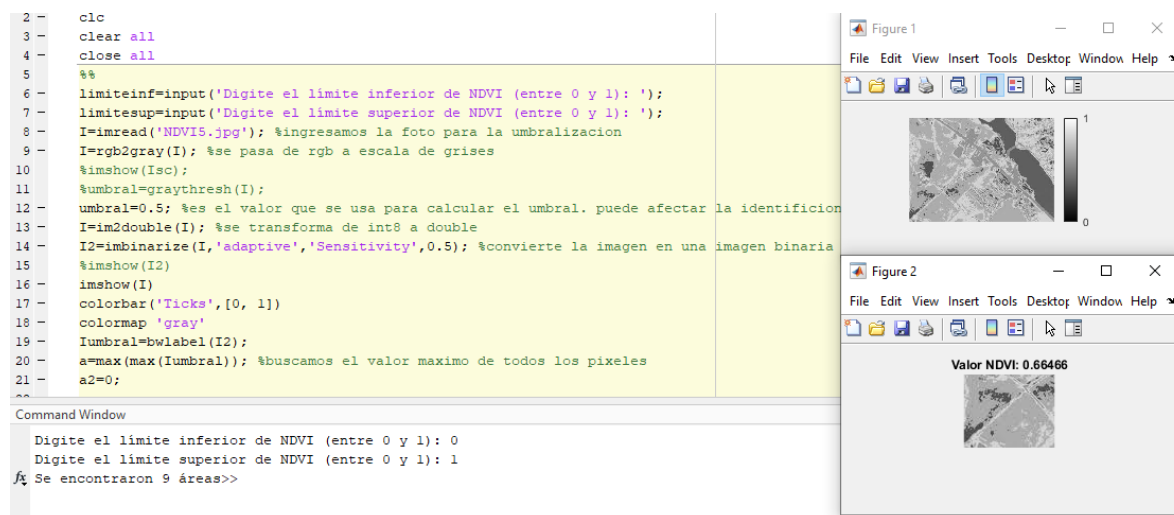
3.1.1 Fase 1: Algoritmos de segmentación basados en procesamiento de imágenes.

Como primer paso, se estudió acerca del procesamiento de imágenes y se escogieron algoritmos claves que conllevarían a la elaboración de nuestro Software propio en App Designer de Matlab. Los algoritmos escogidos implementados fueron:

1. Umbralización por áreas rectangulares, por pixel y la distancia de Mahalanobis; estos tres para la implementación del modelo de productividad, ahorro y gasto basado en imágenes NDVI.
2. Se implementaron algoritmos de detección de contornos entre los cuales encontramos los basados en Laplacianos, Gradiente y Transformada de Hough, y un algoritmo de segmentación basado en k-means, con fines de procesamiento, visualización y extracción de características.

Como paso siguiente los algoritmos descritos en los numerales 1 y 2 se implementaron en Matlab (véase figura 5), antes de ser incorporados en una interfaz gráfica.

Figura 5. Autoría propia [2020]. Códigos base de procesamiento de imágenes en Matlab.



3.1.2 Fase 2: Software propio: interfaces y funcionamiento.

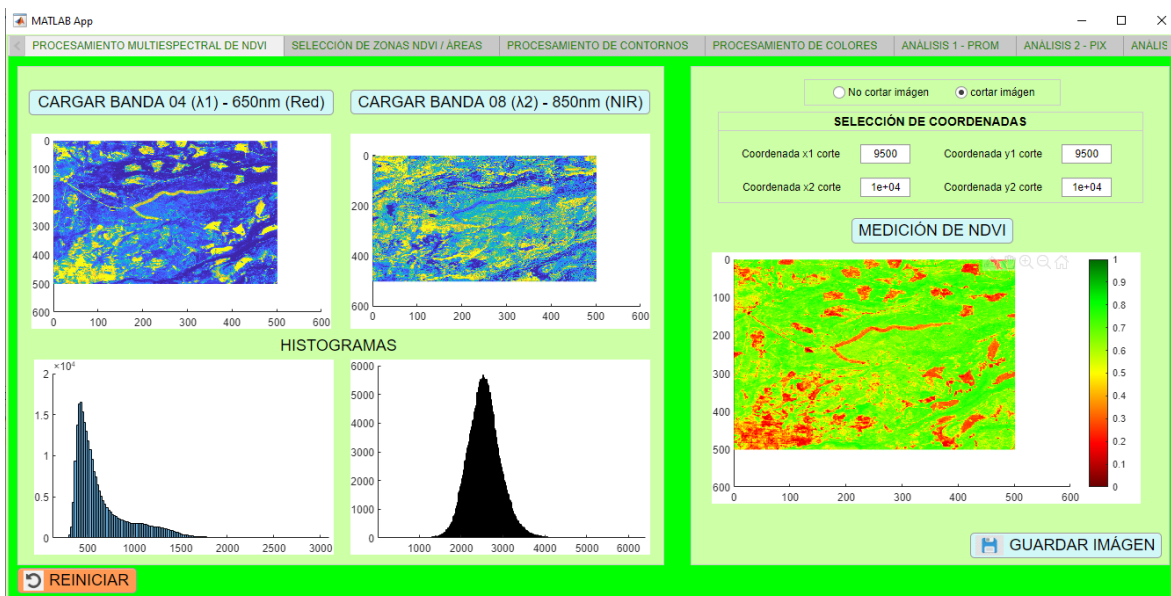
Una vez depurados los algoritmos de la fase 1, se procedió a estudiar la herramienta que brinda Matlab para la creación de aplicaciones: “AppDesigner”. Para lograr esto se diseñaron prototipos de interfaces de usuario como la requerida para este proyecto con el fin de afianzar el manejo de esta herramienta.

Luego con los conocimientos adquiridos se inició el diseño de la interfaz final, con el objetivo de que visualmente fuera amigable a un usuario final, en el cual se incorporarían los algoritmos mencionados anteriormente. Estos se dividieron en diferentes pestañas que se clasificaron según el uso específico, para lo cual resultó una interfaz en 7 pestañas.

- Pestaña #1 (Procesamiento multiespectral de NDVI)

Muestra una interfaz que se usa para cargar imágenes provenientes de las bandas del satélite Sentinel: (banda 04-650 nm y banda 08-850 nm) para ser procesadas y así obtener la imagen NDVI resultante como se observa en el lado derecho de la figura. En esta interfaz se agregan unos histogramas para analizar la uniformidad de estas (véase figura 6).

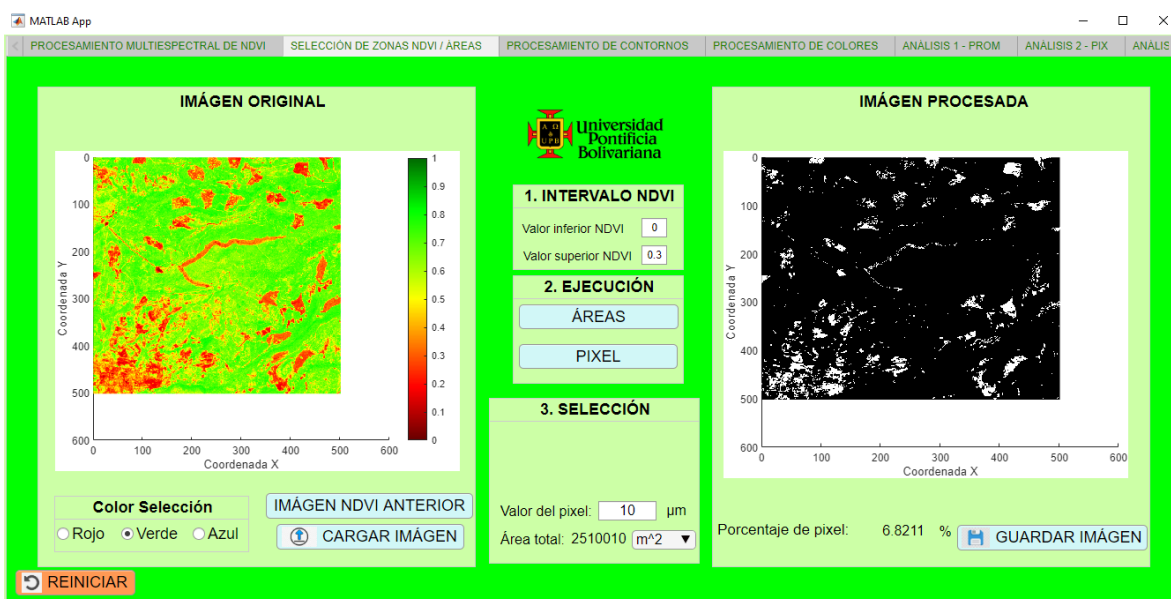
Figura 6. Autoría propia [2020]. Interfaz para cargar imágenes a procesar.



- Pestaña #2 (Selección zonas NDVI/Áreas)

Muestra una interfaz que se usa para cargar la imagen la cual va a ser procesada, esta consta de un panel el cual se le van a ingresar los valores límites de NDVI a hallar y otro donde se selecciona el método de procesamiento a usar (áreas o pixel), y a su derecha se encuentra la imagen procesada resultante (véase figura 7).

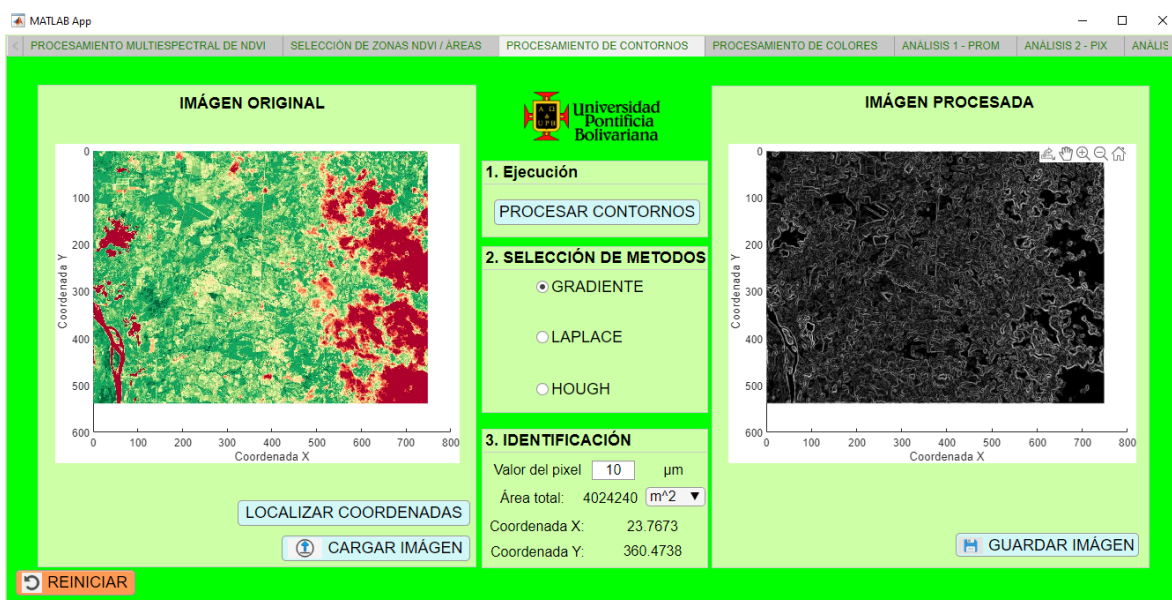
Figura 7. Autoría propia [2020]. Interfaz para la selección de zonas NDVI.



- Pestaña #3 (Procesamiento de contornos)

Muestra una interfaz que se usa para cargar la imagen la cual va a ser procesada, esta consta de un panel en donde se selecciona el método de procesamiento a usar (Gradiente, Laplace o Hough), y a su derecha se encuentra la imagen procesada resultante que muestra los contornos de una imagen (véase figura 8).

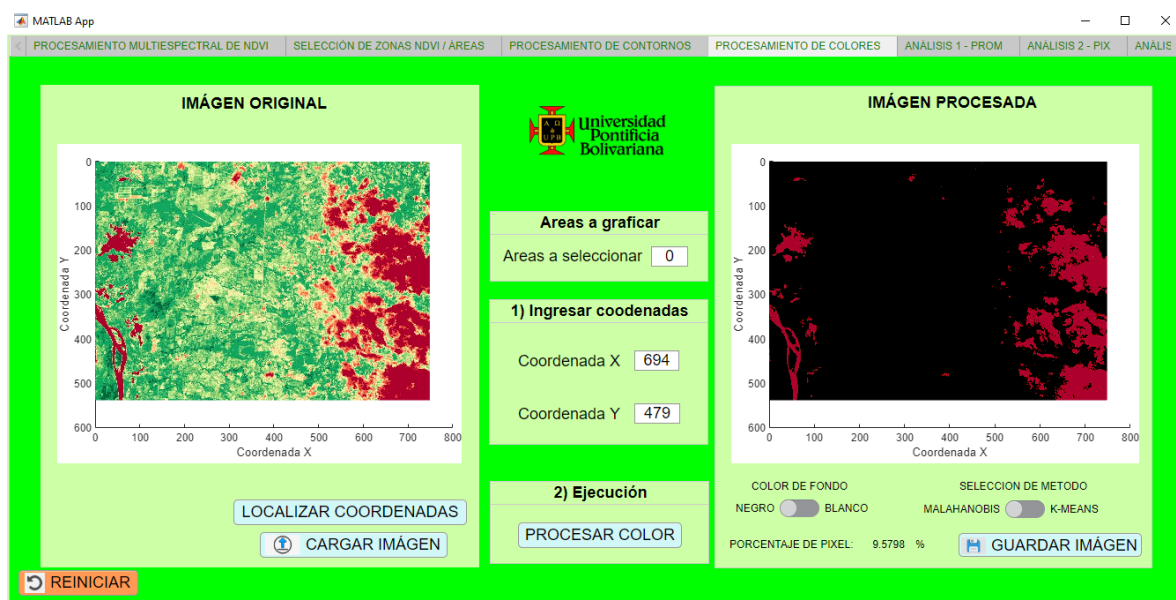
Figura 8. Autoría propia [2020]. Interfaz para procesar contornos.



- Pestaña #4 (Procesamiento de colores)

Muestra una interfaz que se usa para cargar la imagen la cual va a ser procesada, esta consta de un panel en donde se selecciona la coordenada a analizar mediante un clic o ingresando los valores manualmente, y otro donde se selecciona el método de procesamiento a usar (Mahalanobis o Kmeans), y a su derecha se encuentra la imagen procesada resultante (véase figura 9).

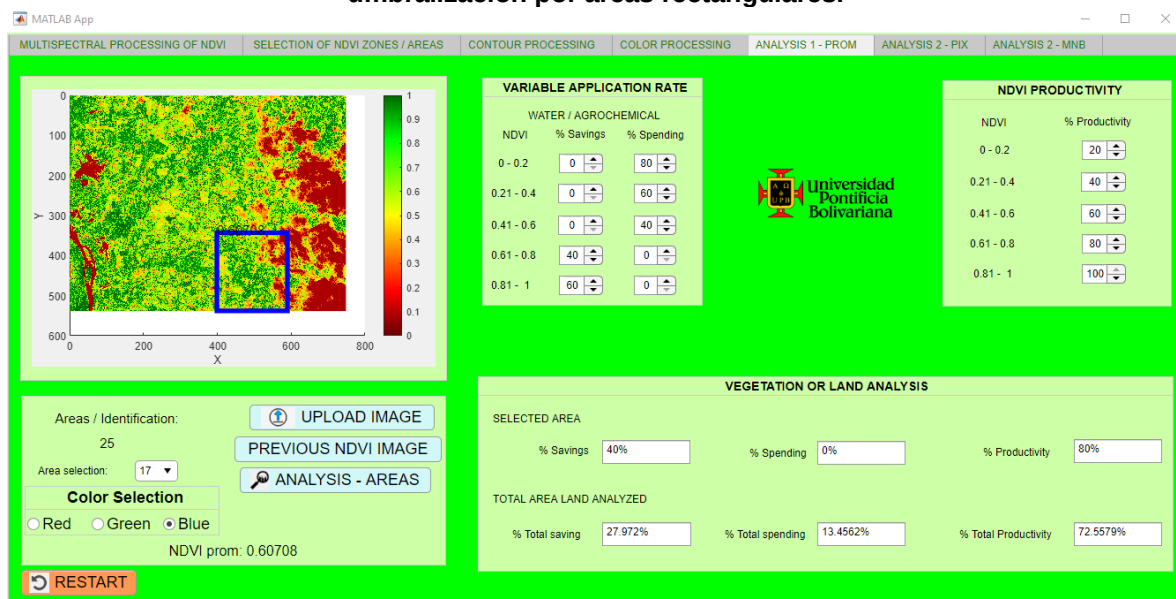
Figura 9. Autoría propia [2020]. Interfaz para procesar color.



- Pestaña #5 (Análisis 1 - PROM)

Muestra una Interfaz para analizar la imagen utilizando el algoritmo de umbralización por áreas rectangulares, esta consta de dos paneles principales en el que se relaciona el modelo lineal para estimación de la productividad de la cosecha, y otro donde se muestran los resultados específicos de cada área seleccionada y resultados totales de la productividad del terreno analizado (véase figura 10).

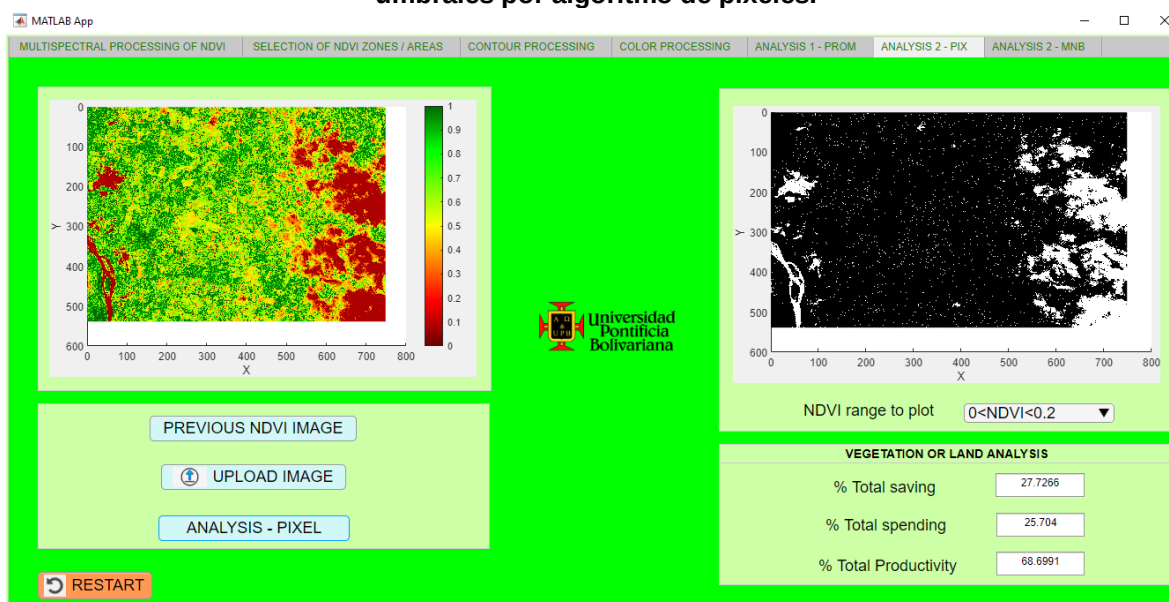
Figura 10. Autoría propia [2020]. Interfaz para procesar imágenes usando el algoritmo de umbralización por áreas rectangulares.



- Pestaña #6 (Análisis 2 - PIX)

Muestra una Interfaz para analizar la imagen utilizando el algoritmo de umbralización por pixel, esta consta de un panel el cual se debe seleccionar el rango de NDVI a graficar y otro donde se muestran los resultados totales de la productividad del terreno analizado (véase figura 11).

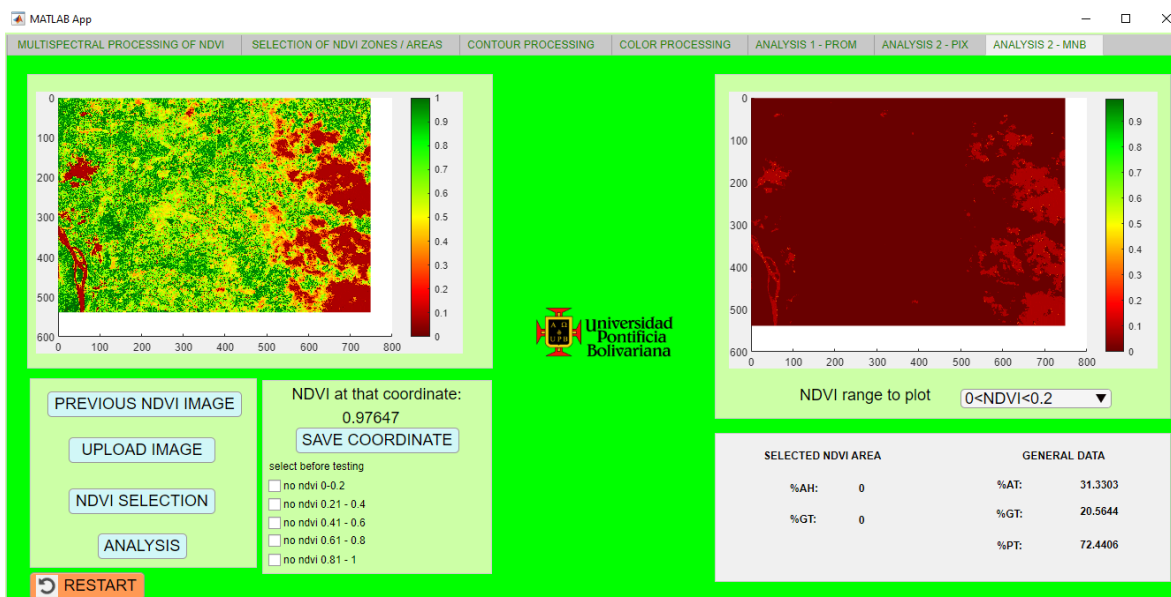
Figura 11. Autoría propia [2020]. Interfaz para procesar imágenes mediante el uso de umbrales por algoritmo de píxeles.



- Pestaña #7 (Análisis 3 - MNB)

Muestra una Interfaz para analizar la imagen utilizando el algoritmo distancia de Mahalanobis esta consta de un panel el cual se debe seleccionar el rango de NDVI a graficar y otro donde muestran los resultados específicos de cada área seleccionada y los resultados totales de la productividad del terreno analizado (véase figura 12).

Figura 12. Autoría propia [2020]. Interfaz para procesar imágenes mediante el uso de distancia de Mahalanobis.

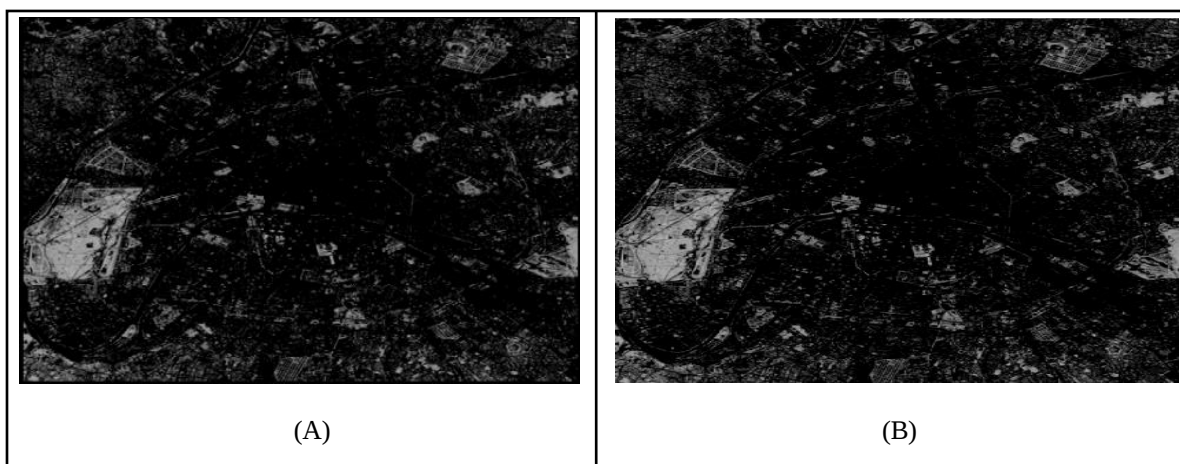


Como anexos están el manual de instalación y manual de funcionamiento.

3.1.3 Fase 3: Comprobación de algoritmos con imágenes satelitales.

Con la interfaz en su fase final, se inició la prueba de algoritmos para conocer si estaban funcionando correctamente. Una de las pruebas realizadas consistió en usar dos imágenes de las bandas infrarrojo y rojo cercano provenientes de la página oficial de Matlab y se comprobó que el algoritmo de detección de NDVI obtenía resultados muy similares a los mostrados en la página anteriormente mencionada (véase figura 13).

Figura 13. Autoría propia [2020]. A) imagen NDVI procesada del software propio vs B) Imagen NDVI página oficial de Matlab.



3.1.4 Fase 4: NDVI y su relación con el modelo lineal de productividad, ahorro y gasto.

Luego de haber comprobado el correcto funcionamiento de los algoritmos de segmentación por Áreas, por píxel y la distancia de Mahalanobis, se decidió implantar un modelo lineal el cual se realizó asignando un porcentaje de productividad y / o valores de ahorro o gasto en función del valor del NDVI definido por la ecuación 10.

Los valores del NDVI están relacionados con la salud de la planta y, por tanto, con la productividad. Los valores de NDVI cercanos a menos de 0,5 representarían una baja productividad, aumentando a alto si este valor está cerca de 1. La productividad se asigna utilizando estos valores:

Tabla 1. Productividad asignada (en porcentaje) según modelo lineal.

Valores NDVI:	0-0.2	0.21-0.4	0.41-0.6	0.61-0.8	0.81-1
Productividad asignada	20%	40%	60%	80%	100%

Tabla 2. Ahorro asignado (en porcentaje) según modelo lineal.

Valores NDVI:	0-0.2	0.21-0.4	0.41-0.6	0.61-0.8	0.81-1
Productividad asignada	0%	0%	0%	40%	60%

Tabla 3. Gasto asignado (en porcentaje) según modelo lineal.

Valores NDVI:	0-0.2	0.21-0.4	0.41-0.6	0.61-0.8	0.81-1
Productividad asignada	80%	60%	40%	0%	0%

El método propuesto para estimar la productividad total, ahorro y gasto (en porcentaje) calculada en una imagen completa depende de la variabilidad del NDVI con la cual se calcula utilizando la suma ponderada de la imagen total (véase ecuaciones 7-8-9), después de los resultados de la segmentación.

4. RESULTADOS Y DISCUSIÓN.

4.1 IMÁGENES SATELITALES UTILIZADAS PARA EL ANÁLISIS.

Como primer paso, se seleccionaron las imágenes satelitales para probar el desempeño del modelo de productividad lineal, ahorro y gasto, y se eligió el valor adecuado, según su dependencia con el valor NDVI (véase tablas 1,2,3) para aplicar los algoritmos de segmentación y estimar los resultados de cada variable agrícola (ahorro, gasto y productividad) en valores porcentuales.

Figura 14. Autoría propia [2020]. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 50 m / píxel con alto contenido de rojo.

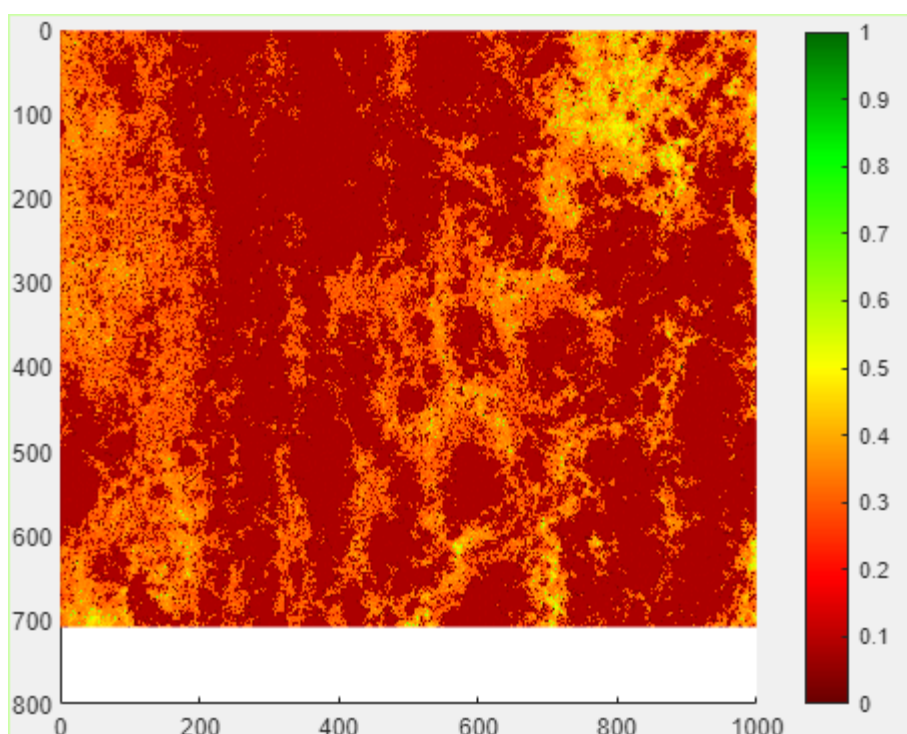


Figura 15. Autoría propia [2020]. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 40 m / píxel con alto contenido de rojo.

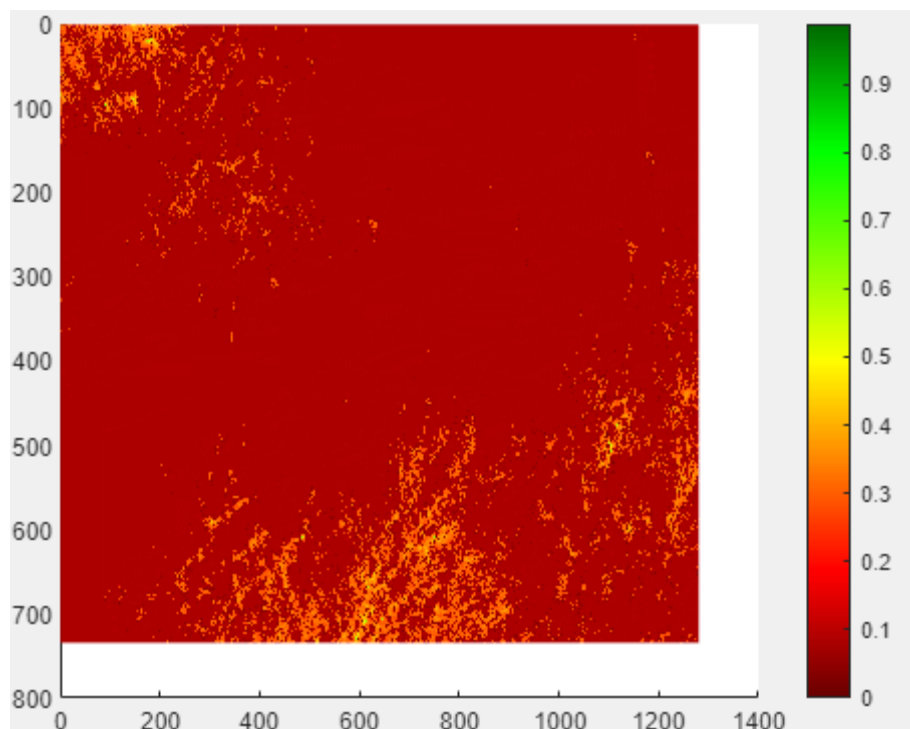


Figura 16. Autoría propia [2020]. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 40 m / píxel con alto contenido de amarillo.

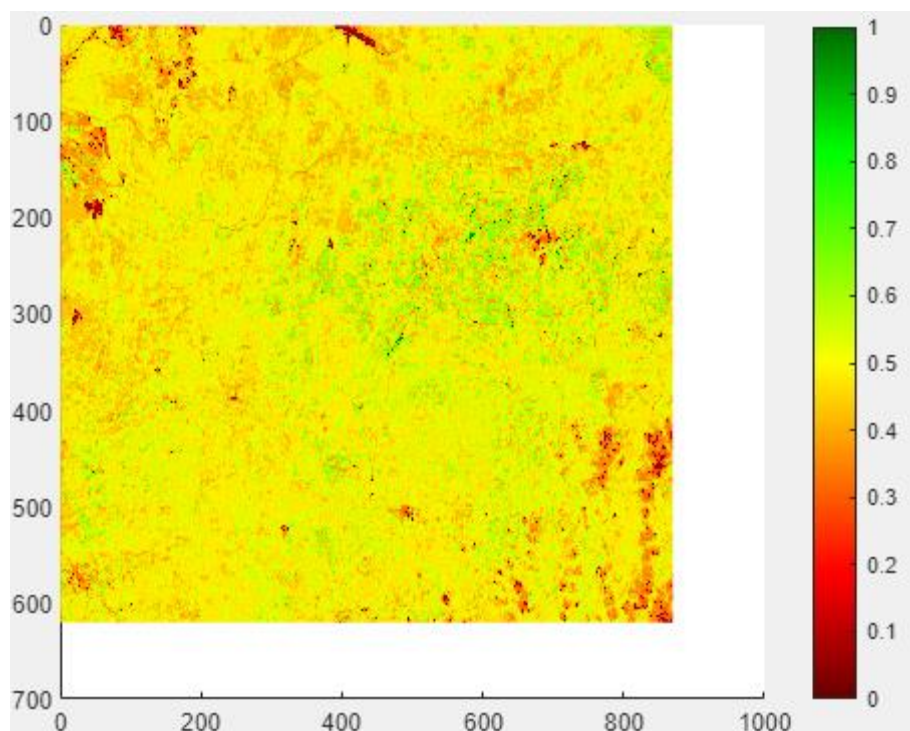


Figura 17. Autoría propia [2020]. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 40 m / píxel con alto contenido de amarillo.

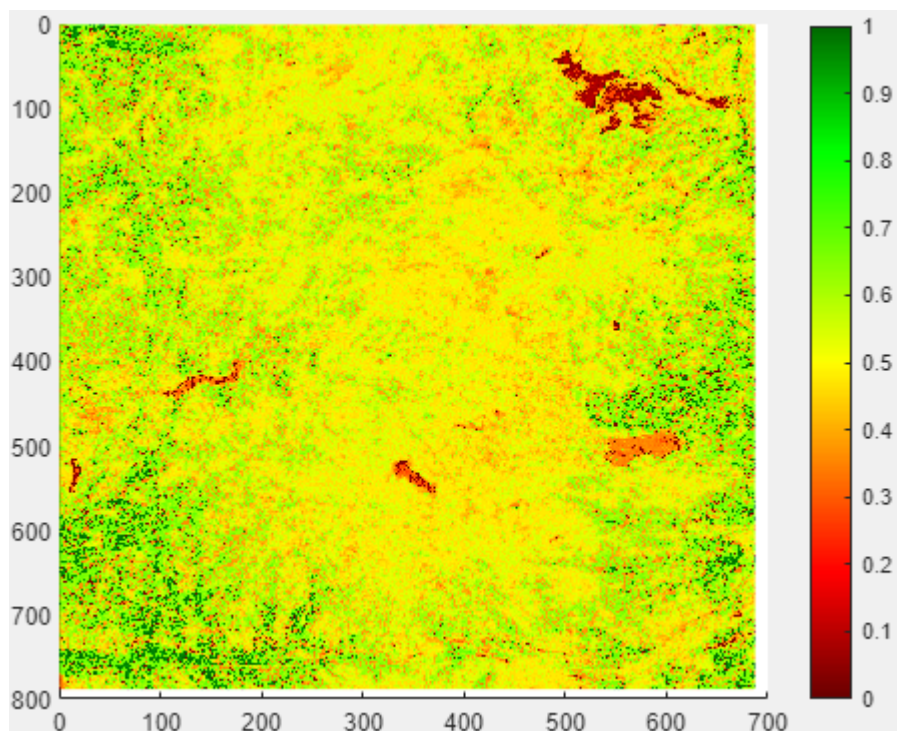


Figura 18. Autoría propia [2020]. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 40 m / píxel con alto contenido de verde.

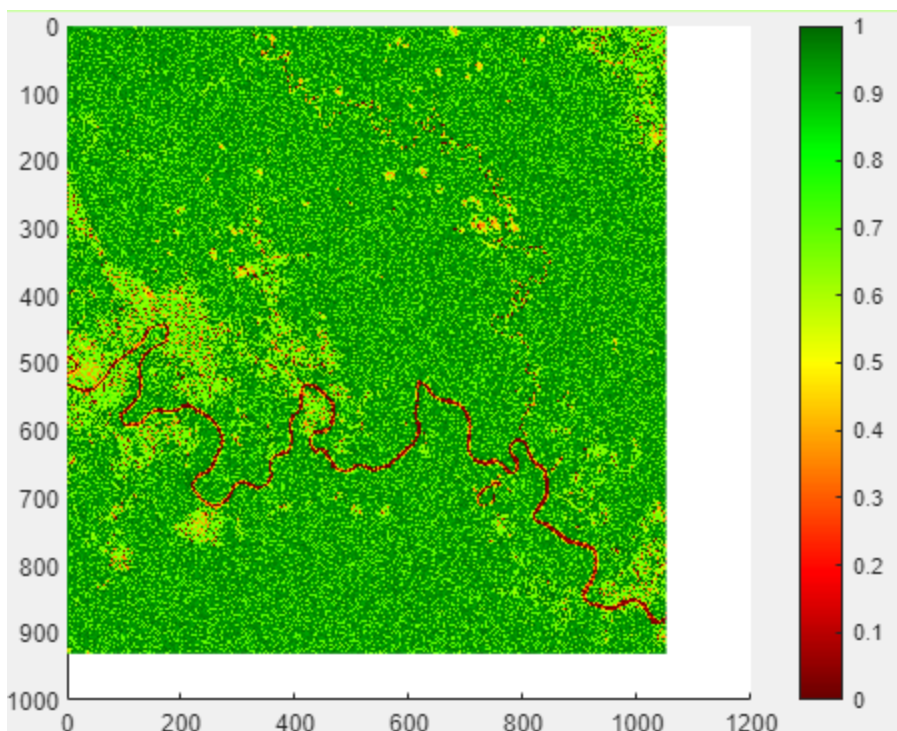


Figura 19. Autoría propia [2020]. Imagen NDVI para probar el modelo lineal de productividad de rendimiento en imágenes ligeramente no uniformes con resolución de 40 m / píxel con alto contenido de verde.

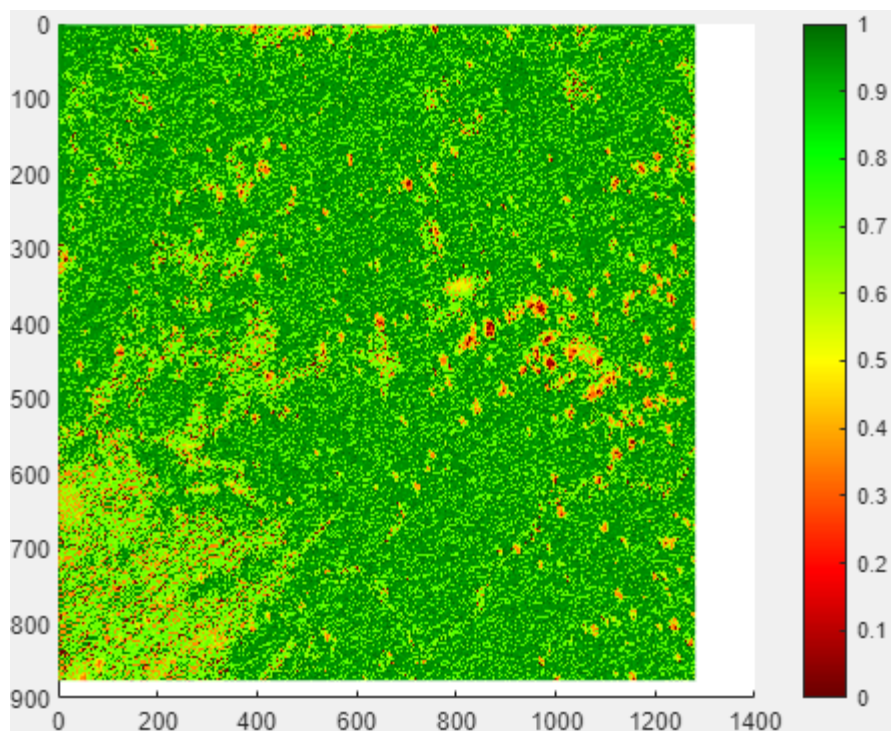


Figura 20. Autoría propia [2020]. Imágenes NDVI para probar el modelo lineal de productividad de rendimiento en imágenes altamente no uniformes con resolución de 50 m / píxel.

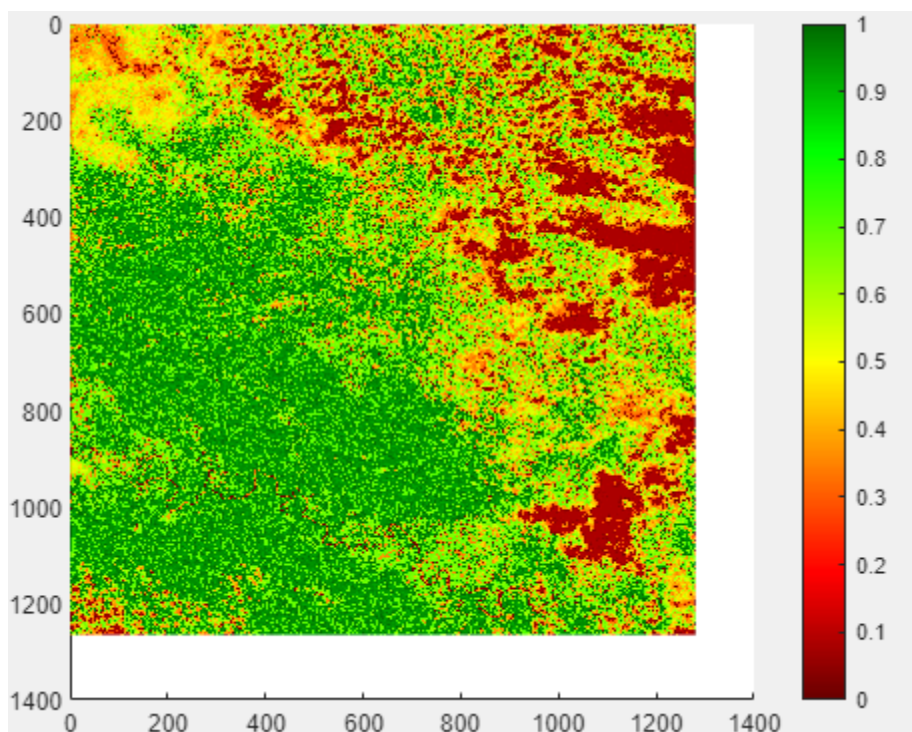


Figura 21. Autoría propia [2020]. Imágenes NDVI para probar el modelo lineal de productividad de rendimiento en imágenes altamente no uniformes con resolución de 40 m / píxel.

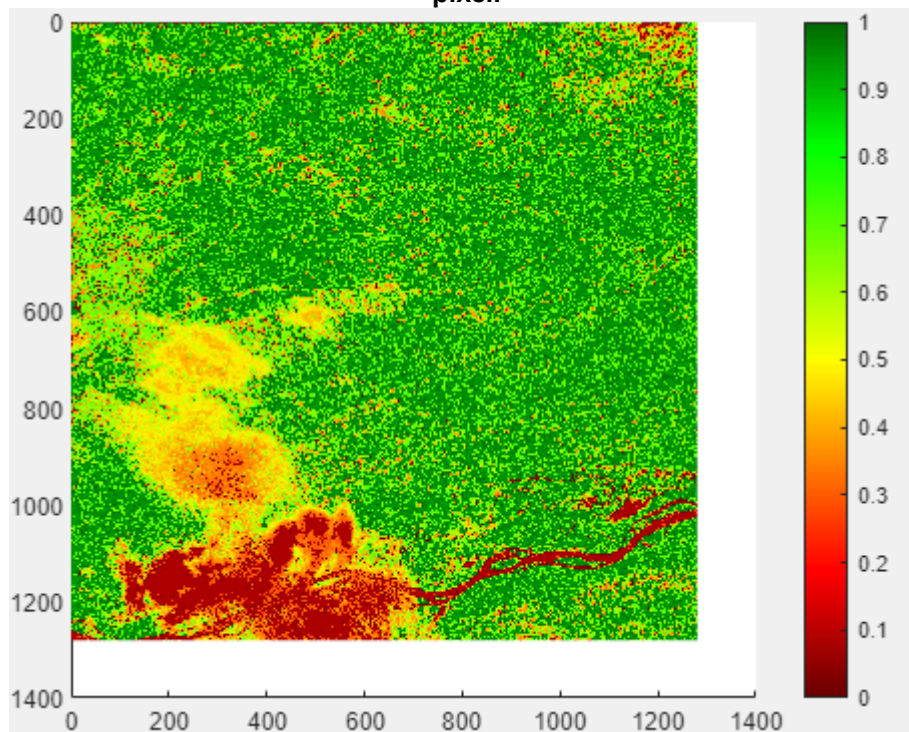
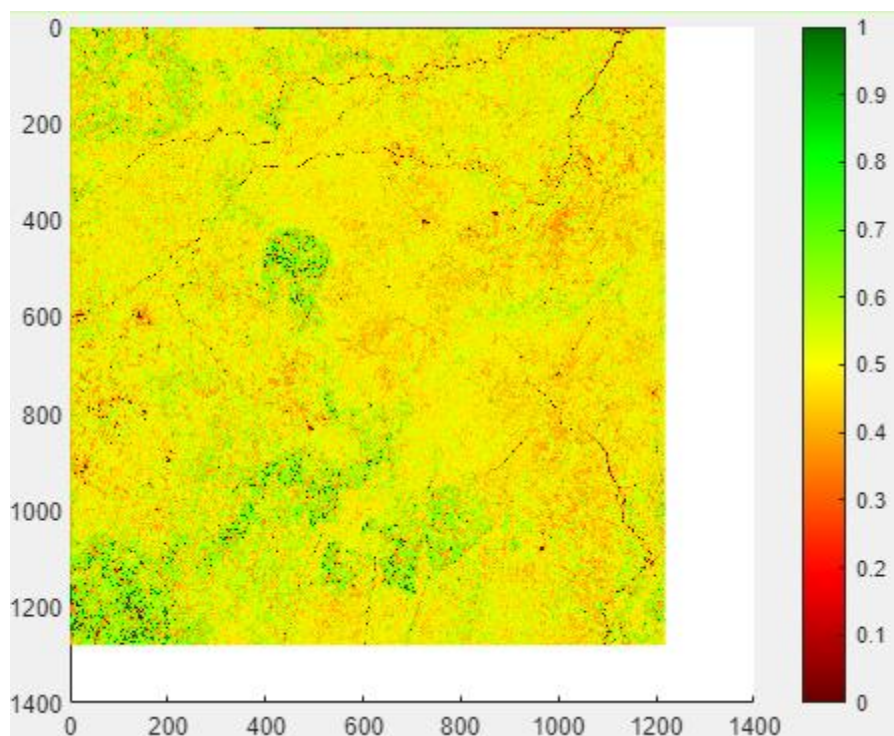


Figura 22. Autoría propia [2020]. Imágenes NDVI para probar el modelo lineal de productividad de rendimiento en imágenes altamente no uniformes con resolución de 40 m / píxel.



Con el fin de probar nuestro modelo lineal y su sensibilidad a las desviaciones pequeñas de NDVI, se seleccionaron imágenes ajustándose al criterio de uniformidad dado por su histograma de frecuencias. En un marco ideal, se probarían imágenes con todo el contenido de rojo, amarillo o verde, y se esperarían valores de productividad cercanos, después de ejecutar los tres algoritmos de segmentación. Sin embargo, el objetivo principal al utilizar imágenes destacadas con una ligera falta de uniformidad es probar la sensibilidad a una pequeña variación de NDVI de la imagen, que se puede observar cómo píxeles con una ligera desviación del valor principal deseado, ya sea rojo, amarillo o verde. Entonces, estas imágenes se consideran mayormente rojas, amarillas o verdes, pero no totalmente.

Usando los algoritmos de segmentación mencionados anteriormente, el valor de las variables agrícolas que se correlacionan con la distribución de NDVI, se espera que sea el correspondiente al asignado previamente en la Tabla 1, 2 y 3. Las Imágenes con alto contenido de rojo (véase figura 14 y 15) se espera un valor de productividad cercano al 20%, valor de ahorro cercano al 0% y de gasto cercano a un 80%. Las Imágenes con alto contenido de amarillo (véase figura 16 y 17) se espera un valor de productividad cercano al 60%, valor de ahorro cercano al 0% y de gasto cercano a un 40%. Y las imágenes con alto contenido de verde (véase figura 18 y 19) se espera un valor de productividad cercano al 100%, valor de ahorro cercano al 60% y de gasto cercano a un 0%. A partir de esta prueba inicial se puede comprobar que nuestro algoritmo está “calibrado” a las asignaciones previas de las tablas en mención.

4.2 ESTIMACIÓN DE LA PRODUCTIVIDAD UTILIZANDO IMÁGENES LIGERAMENTE NO UNIFORMES.

Se escoge como el algoritmo de referencia “umbralización por pixel”, como valor más confiable, ya que este realiza un análisis que como su nombre lo indica, examina pixel por pixel en cada valor de la matriz y si este valor no está dentro del umbral lo vuelve 0, lo que quiere decir que tiene un rango de tolerancia de 0% por lo que es más confiable que los demás códigos de segmentación. La productividad, ahorro y gasto (en porcentaje) calculados en cada una de las imágenes ligeramente no uniformes (Figuras 14-19) se muestran en las tablas 4, 5 y 6 respectivamente. Usando este tipo de imágenes con distribución NDVI no uniforme, el algoritmo de umbralización por áreas promedia los valores NDVI aproximándose al valor del píxel principal de la imagen con el valor NDVI correspondiente, por ejemplo, al rojo (0,2), amarillo (0,6) y verde (1). Mientras que el algoritmo de Mahalanobis realiza estos cálculos utilizando su respectiva fórmula matemática (ecuación 6), y el algoritmo de píxel realiza este proceso mediante la aplicación de umbrales límites sin rango de tolerancia. Después de este proceso se obtiene una matriz de valores NDVI y se le aplica su valor de productividad previamente asignado según el color: rojo (20%), amarillo (60%) y verde (100%). En este caso, es más evidente el efecto de la no uniformidad introducida de las imágenes. Se concluye que, los algoritmos de segmentación de umbralización por píxel y la distancia de Mahalanobis son más

sensibles para detectar la falta de uniformidad en la distribución de píxeles del NDVI en las imágenes, y operar con mejor precisión.

Tabla 4. Valores de productividad estimados mediante modelo lineal, segmentación por procesamiento de imágenes en imágenes de la Figura 14-19.

Imagen de la figura 14	
Algoritmos	% Productividad
Umbralización por áreas	20
Umbralización por pixel	21.4
Distancia de Mahalanobis	22
Imagen de la figura 15	
Algoritmos	% Productividad
Umbralización por áreas	20.20
Umbralización por pixel	27.75
Distancia de Mahalanobis	28.11
Imagen de la figura 16	
Algoritmos	% Productividad
Umbralización por áreas	60
Umbralización por pixel	59.25
Distancia de Mahalanobis	66.49
Imagen de la figura 17	
Algoritmos	% Productividad
Umbralización por áreas	60.71
Umbralización por pixel	62.21
Distancia de Mahalanobis	63.61
Imagen de la figura 18	
Algoritmos	% Productividad
Umbralización por áreas	99.9
Umbralización por pixel	91.95
Distancia de Mahalanobis	97.95
Imagen de la figura 19	
Algoritmos	% Productividad
Umbralización por áreas	99.51
Umbralización por pixel	90.18
Distancia de Mahalanobis	98.7

Tabla 5. Valores de ahorro estimados mediante modelo lineal, segmentación por procesamiento de imágenes de la Figura 14-19

Imagen de la figura 14	
Algoritmos	% Ahorro
Umbralización por áreas	0
Umbralización por pixel	0
Distancia de Mahalanobis	0
Imagen de la figura 15	
Algoritmos	% Ahorro
Umbralización por áreas	0
Umbralización por pixel	0
Distancia de Mahalanobis	0
Imagen de la figura 16	
Algoritmos	% Ahorro
Umbralización por áreas	0
Umbralización por pixel	1.19
Distancia de Mahalanobis	3.94
Imagen de la figura 17	
Algoritmos	% Ahorro
Umbralización por áreas	1.42
Umbralización por pixel	8.75
Distancia de Mahalanobis	9.5
Imagen de la figura 18	
Algoritmos	% Ahorro
Umbralización por áreas	59.91
Umbralización por pixel	52.05
Distancia de Mahalanobis	57.2
Imagen de la figura 19	
Algoritmos	% Ahorro
Umbralización por áreas	59.51
Umbralización por pixel	50.08
Distancia de Mahalanobis	56.27

Tabla 6. Valores de gasto estimados mediante modelo lineal, segmentación por procesamiento de imágenes de la Figura 14-19.

Imagen de la figura 14	
Algoritmos	% Gasto
Umbralización por áreas	80
Umbralización por pixel	78.60
Distancia de Mahalanobis	79.92
Imagen de la figura 15	

Algoritmos	% Gasto
Umbralización por áreas	80
Umbralización por pixel	72.20
Distancia de Mahalanobis	71.28
Imagen de la figura 16	
Algoritmos	% Gasto
Umbralización por áreas	40
Umbralización por pixel	40.17
Distancia de Mahalanobis	41.56
Imagen de la figura 17	
Algoritmos	% Gasto
Umbralización por áreas	39
Umbralización por pixel	34.18
Distancia de Mahalanobis	30.6
Imagen de la figura 18	
Algoritmos	% Gasto
Umbralización por áreas	0
Umbralización por pixel	2.7
Distancia de Mahalanobis	1.27
Imagen de la figura 19	
Algoritmos	% Gasto
Umbralización por áreas	0
Umbralización por pixel	3.78
Distancia de Mahalanobis	1.73

4.3 ESTIMACIÓN DE LA PRODUCTIVIDAD MEDIANTE IMÁGENES NO MUY UNIFORMES.

Como segundo paso, se eligieron imágenes (ver figuras 20,21 y 22) para probar el desempeño del modelo de productividad lineal, ahorro y gasto (en porcentaje) y su dependencia con los valores NDVI utilizando imágenes altamente no uniformes para aplicar los algoritmos de segmentación y estimar el resultado de estos.

La productividad, ahorro y gasto (porcentaje) calculados en cada una de las imágenes no muy uniformes se muestran en las Tabla 7,8 y 9. Utilizando este tipo de imágenes con distribución NDVI no uniforme, el algoritmo de umbralización por píxel, distancia de Mahalanobis y áreas muestran una leve diferencia con desviaciones de máximo 13%. En este caso es más evidente el efecto debido a la introducción de una gran falta de uniformidad en las imágenes. Como se mencionó en la subsección anterior, se puede concluir que el umbral por píxel y la distancia de Mahalanobis son más adecuados para segmentar y calcular la productividad en una imagen completa, mostrando una mejor precisión. Tomando como criterio de

precisión la cercanía o la desviación porcentual considerando el valor arrojado por el algoritmo de referencia: algoritmo de umbralización por pixel.

Tabla 7. Valores de productividad estimados mediante el modelo lineal, segmentación por procesamiento de imágenes en imágenes de la figura 20-22.

Imagen de la figura 20	
Algoritmos	% Productividad
Umbralización por áreas	79.79
Umbralización por pixel	71.10
Distancia de Mahalanobis	81.32
Imagen de la figura 21	
Algoritmos	% Productividad
Umbralización por áreas	77.01
Umbralización por pixel	81.74
Distancia de Mahalanobis	87.78
Imagen de la figura 22	
Algoritmos	% Productividad
Umbralización por áreas	60
Umbralización por pixel	60.28
Distancia de Mahalanobis	64.51

Tabla 8. Valores de ahorro estimados mediante el modelo lineal, segmentación por procesamiento de imágenes en imágenes de la figura 20-22.

Imagen de la figura 20	
Algoritmos	% Ahorro
Umbralización por áreas	39.67
Umbralización por pixel	30.71
Distancia de Mahalanobis	35.17
Imagen de la figura 21	
Algoritmos	% Ahorro
Umbralización por áreas	34.89
Umbralización por pixel	41.34
Distancia de Mahalanobis	46.61
Imagen de la figura 22	
Algoritmos	% Ahorro
Umbralización por áreas	0
Umbralización por pixel	3.37
Distancia de Mahalanobis	2.33

Tabla 9. Valores de gasto estimados mediante el modelo lineal, segmentación por procesamiento de imágenes en imágenes de la figura 20-22.

Imagen de la figura 20	
Algoritmos	% Gasto
Umbralización por áreas	0
Umbralización por pixel	23.55
Distancia de Mahalanobis	22.66
Imagen de la figura 21	
Algoritmos	% Gasto
Umbralización por áreas	6
Umbralización por pixel	13.08
Distancia de Mahalanobis	9.77
Imagen de la figura 22	
Algoritmos	% Gasto
Umbralización por áreas	40
Umbralización por pixel	38.24
Distancia de Mahalanobis	44.44

5. CONCLUSIONES Y RECOMENDACIONES.

En este trabajo se desarrolló e implementó un software de procesamiento de imágenes que consta de 7 pestañas que cumplen labores de segmentación usando los métodos de umbralización por (áreas rectangulares, pixel y distancia de Mahalanobis), procesamiento de colores usando los métodos de distancia de Mahalanobis y k-means y procesamiento de contornos usando los métodos de Hough, Gradiente y Laplaciano. A partir de estos algoritmos se obtuvieron resultados de variables agrícolas tales como: la productividad, ahorro y gasto de la cosecha mediante la asignación previa de un modelo lineal en función del NDVI y su distribución.

Los resultados muestran que estos métodos de segmentación se pueden utilizar para estimar la productividad, o en general en proyectos de Agricultura 4.0 de acuerdo con las necesidades particulares de los agricultores.

El método de umbralización por píxeles se seleccionó como método de referencia y como el más exacto de todos los métodos presentados para la estimación de variables agrícolas, tanto para imágenes ligeramente uniforme como las que no presentan uniformidad. Además, es el que menos carga computacional tiene por lo que lo convertiría en el más factible para ser implementado en diferentes tipos de proyectos que conlleven procesamiento de imágenes, por ejemplo, en un sistema de Inteligencia de Artificial basado en técnicas de Machine Learning.

Este método a su vez es utilizado como un dato teórico para el análisis de errores para la estimación de la productividad, ahorro y gasto del cultivo comparando ese valor con los calculados con los otros algoritmos (umbralización por área y Mahalanobis). Umbralización por pixel hace un análisis de cada una de las posiciones de la matriz de datos de la imagen, mientras que los algoritmos de segmentación por áreas y por distancia de Mahalanobis pueden presentar variaciones dependiendo de la uniformidad de las imágenes utilizadas.

Para el caso de segmentación por áreas, presenta un promedio de rangos de NDVI previamente solicitado por el usuario, puede generar más desviaciones en los resultados en imágenes no muy uniformes. Alternativamente, el método de segmentación por la distancia de Mahalanobis el cual se basa en una fórmula matemática que depende de desviaciones con respecto a una distancia, que a su vez depende mucho del valor de NDVI ingresado por el usuario en cada rango propuesto, por ejemplo, en el rango de 0.4-0.6 el resultado será diferente si se escoge un valor de 0.48 a si se escoge un valor de 0.56, esta diferencia será mayor si se escogen valores extremos en el rango. Al tener un permitido de 0.2 en cada intervalo, se puede tomar cualquier valor decimal dentro de ese, por lo que tiende a arrojar diferentes resultados de imágenes segmentadas, al ser muy sensible respecto al valor ingresado.

Como se pueden observar en las tablas obtenidas (véase tablas 4-9), aún con lo

mencionado anteriormente, todos estos algoritmos de segmentación (umbralización por áreas rectangulares, por pixel y la distancia de Mahalanobis) trabajan con una desviación inferior al 13%, y se pueden utilizar para gestionar de forma diferencial áreas segmentadas en términos de cualquier variable agrícola elegida como el ahorro, el gasto de recursos o la productividad.

En caso de que se desee seguir desarrollando este software, se plantean estas posibles mejoras:

- Independizar el procesamiento de imágenes RGB del Multiespectrales para discriminación de biomasa con el suelo.
- Adaptar el software para que pueda realizar el procesamiento de imágenes con diferentes longitudes de onda usando cámaras hyperspectrales.
- Técnicas de segmentación semántica, la cual consiste en un método de Deep learning que permite reconocer características dentro de los píxeles de la imagen.

6. BIBLIOGRAFÍA

- Adam, Y., Amaya, M., & Melendez, J. (23 de enero de 2012). *Aplicaciones de la transformada de Fourier*. Obtenido de issuu: https://issuu.com/jenniffer2012/docs/aplicaciones_de_la_transformada_de_Fourier.
- Alonso, D. (s.f.). *NDVI: Qué es y cómo calcularlo con SAGA desde QGIS*. Obtenido de mappingGIS: https://mappinggis.com/2015/06/ndvi-que-es-y-como-calcularlo-con-saga-desde-qgis/#Que_es_un_NDVI
- Arellano, R. (2014). El espectro de frecuencias y sus aplicaciones. *asdopen*, 6-7.
- Bhattarai, R., Bajgiran, P. R., Weiskittel, A., & MacLean, D. A. (2020). Sentinel-2 based prediction of spruce budworm defoliation using red-edge spectral vegetation indices. *Remote Sensing Letters DOI: 10.1080/2150704X.2020.1767824*.
- Candiago, S., Remondino, F., de Giglio, M., Dubbini, M., & Gattelli, M. (2015). Evaluating Multispectral Images and Vegetation Indices for Precision Farming Applications from UAV Images. *Remote Sens. 2015*, 7(4), 4026-4047; <https://doi.org/10.3390/rs70404026>.
- Cárdenes, R., & Ruiz Alzola, J. (Noviembre de 2005). *Segmentación de Tejidos Cerebrales Mediante Transformadas de Distancias con la Métrica de Mahalanobis*. Obtenido de ResearchGate.
- Carrillo, M. A. (2011). *Procesamiento Digital de Señales*.
- Choudhary, K., Shi, W., Boori, M. S., & Corgne, S. (2019). Agriculture Phenology Monitoring Using NDVI Time Series Based on Remote Sensing Satellites: A Case Study of Guangdong, China. *Opt. Mem. Neural Networks 28*, 204–214 *Doi:10.3103/S1060992X19030093*.
- DIFERENCIA NORMALIZADA DE BORDE ROJO. (s.f.). Obtenido de EARTH OBSERVING SYSTEM: <https://eos.com/agriculture/ndre/es>
- Dongdong, M., Liangju, W., Libo, Z., Tanzeel U., R., JIAN, J., & Yunhong, D. (2020). Precise Estimation of NDVI with a Simple NIR Sensitive RGB Camera and Machine Learning Methods for Corn Plants. *DOI: 10.3390/s20113208*.
- Dongdong, M., Liangju, W., Libo, Z., Zhihang, S., Tanzeel U., R., & JIAN, J. (2020). Stress Distribution Analysis on Hyperspectral Corn Leaf Images for Improved Phenotyping Quality. 8-12.
- dos Santos, M. R., Madalozzo, G. A., Cunha Fernandes, J. M., & Rieder, R. (2020). A Computer Vision System for High-Throughput Phenotyping. *IJAEIS 11.1 1-22 Doi:10.4018/IJAEIS.2020010101*.
- Escalera, A. d. (2016). *Visión por Computador*.
- García, E., & Flego, F. (s.f.). *Agricultura de precisión*. Obtenido de <https://www.palermo.edu/ingenieria/downloads/pdfwebc&T8/8CyT12.pdf>
- González, F. A., Gómez, J. J., & Amaya, D. F. (2017). Procesamiento de imágenes multiespectrales en cultivos de café y cacao. *Cintex 22 (2)*, 51-67.
- Haggar, R. J., Stent, C. J., & Rose, J. (1984). Measuring spectral differences in vegetation canopies by a reflectance ratio meter. *doi:10.1111/j.1365-3180.1984.tb00571*.

- Hasseb, K., Din , I. U., Almogren, A., & Islam , N. (2020). An Energy Efficient and Secure IoT-Based WSN Framework: An Application to Smart Agriculture. *Sensors* 20(7), 2081; Doi:10.3390/s20072081.
- Hussain, F., & Jeong, J. (10 de noviembre de 2015). *Efficient Deep Neural Network for Digital Image Compression Employing Rectified Linear Neurons*. Obtenido de Hindawi: <https://www.hindawi.com/journals/js/2016/3184840/>
- Insights, A. (s.f.). *¿Qué es y cómo funciona una cámara multiespectral?* Obtenido de Aerial Insights: <https://www.aerial-insights.co/blog/camara-multiespectral>
- Ji, S., Bai, X., Qiao, R., Wang, L., & Chang, X. (2020). Width identification of transition zone between desert and oasis based on NDVI and TCI. Doi:10.1038/s41598-020-65286-5.
- Lara Gutiérrez , H., Lara Ruiz, M. G., Hernández Hernández, V., Hernández Hernández, B., & Hernández Hernández , G. (2016). *ANÁLISIS DE UN CASO PRÁCTICO APLICANDO EL ALGORITMO K MEANS MEDIANTE WEKA (WAIKATO ENVIRONMENT FOR KNOWLEDGE ANALYSIS)*. Obtenido de <https://repository.uaeh.edu.mx/revistas/index.php/huejutla/article/download/1135/4718?inline=1>
- Li , W., Clark, B., Taylor, J. A., Kendall, H., Jones , G., Li, Z., . . . Frewer, L. J. (2020). A hybrid modelling approach to understanding adoption of precision agriculture technologies in Chinese cropping systems. *ScienceDirect* Doi:10.1016/j.compag.2020.105305.
- Maccioni, A., Agati, G., & Mazzinghi, P. (2001). New vegetation indices for remote measurement of chlorophylls based on leaf directional reflectance spectra. *Revista de Fotoquímica y Fotobiología B: Biología*, 52-61.
- mesurex. (s.f.). *mesurex*. Obtenido de Cámaras Hiperespectrales: <https://www.mesurex.com/product-category/camaras-hiperespectrales>
- Moreira, L., Castro, F., Góes, J., Bins , L., Teruel , B., Fracarolli, J., . . . Hernandez , F. (s.f.). *H. Applications* . 2-7.
- Moreno, W. F., Tangarife , H. I., & Díaz , A. E. (23 de junio de 2017). *Aplicaciones de análisis de imágenes en agricultura de precisión*. Obtenido de Ebscohost: <https://web-b-ebshost-com.consultaremoti.upb.edu.co/ehost/pdfviewer/pdfviewer?vid=4&sid=85f40647-42ae-4884-bdce-a26c8048ef52%40pdc-v-sessmgr02>
- Moreno, W. F., Tangarife, H. I., & Díaz, A. E. (2017). *Aplicaciones de análisis de imágenes en agricultura de precisión*.
- Mukesh Singh, B., Komal, C., & Victorovich, K. A. (2020). Crop growth monitoring through Sentinel and Landsat data based NDVI time-series. *pp. 409–419*. doi:10.18287/2412-6179-CO-635.
- Peña, L., Rentería , V., Velásquez , C., Ojeada , M., & Barrera , E. (2019). Absorbancia y reflectancia de hojas de Ficus contaminadas con nanopartículas de plata. *Revista Mexicana de física* .
- Puranik, P., P, B., & Palsodkar, P. (2009). *Fuzzy based Color Image Segmentation using Comprehensive Learning Particle Swarm Optimization (CLPSO) -- A Design Approach*. Obtenido de Ebscohost: [https://web-b-ebshost-](https://web-b-ebshost-com.consultaremoti.upb.edu.co/ehost/pdfviewer/pdfviewer?vid=4&sid=85f40647-42ae-4884-bdce-a26c8048ef52%40pdc-v-sessmgr02)

- com.consultaremota.upb.edu.co/ehost/pdfviewer/pdfviewer?vid=7&sid=30ee5dd8-5361-4804-839c-0a887296c4a9%40pdc-v-sessmgr06
- Sánchez, G. S. (2009). *Medición del ruido en la línea de transmisión eléctrica generado por electrodomésticos, para aplicaciones de PLC*. Cholula.
- Soliman, S., & Srinath, M. (1999). *Señales y sistemas continuos y discretos*. Madrid: Prentice Hall.
- Triana, N., Jaramillo, A., Gutiérrez, R., & Rodríguez, C. (2016). *Técnicas de umbralización para el procesamiento de imágenes de GEM-Foils*.
- Urrea, J. P., & Ospina, E. (2004). *IMPLEMENTACIÓN DE LA TRANSFORMADA DE HOUGH PARA LA DETECCIÓN*.

7. ANEXOS.

Adjuntos a este documento se encuentran los siguientes anexos:

- Manuscrito Artículo.
- Código fuente.
- Manual de instalación.
- Manual de operaciones.
- Prueba de funcionamiento. (documento Word)
- Prueba de funcionamiento. (Video de evidencia)
- Registro de Software ante el DNDA (Dirección Nacional de Derechos de Autor)
- Certificado de Participación en Ponencia