Methods for estimating agricultural cropland yield based on the comparison of NDVI images analyzed by means of Image segmentation algorithms: A tool for spatial planning decisions

Métodos de estimación del rendimiento de las tierras de cultivo basados en la comparación de imágenes NDVI analizadas mediante algoritmos de segmentación de imágenes: Una herramienta para la toma de decisiones de planificación espacial

David D. Hernández Molina[1] https://orcid.org/0000-0003-3189-3019
Julio M. Gulfo Galaraga[1] https://orcid.org/0000-0002-9340-4033
Ana Milena López López[1] https://orcid.org/0000-0002-0118-8434
Claudia M. Serpa Imbett[2*] https://orcid.org/0000-0002-1420-0084

Recibido 23 de noviembre de 2022, aceptado 31 de julio de 2023
Received: November 23, 2022 Accepted: July 31, 2023

ABSTRACT

This research study compares the performance of different digital image processing algorithms based on computer vision segmentation methods to process satellite multispectral images of Normalized Difference Vegetation Index (NDVI) to estimate agricultural cropland yield as a proposal for supporting spatial planning decisions. NDVI multispectral images were collected from Sentinel-2 L2-A satellite with distinctive features to be processed through these algorithms implemented in an owned and friendly software interface developed in MATLAB App Designer. These are based on image color detection, using three techniques: rectangular thresholding method, simple thresholding method, and segmentation through Mahalanobis discriminant classifier. The segmented images were used to estimate cropland yields as a function of NDVI variations and the characteristics of each analyzed image, employing a linear model that assigned a yield to each segmented area as a function of a specific NDVI range. Algorithm accuracy was determined as a function of expected cropland yield. Results show that the rectangular thresholding method tends to average cropland yield value in slightly non-uniform images. In contrast, thresholding by pixel and Mahalanobis methods performed better on highly non-uniform NDVI images, with deviations less than 8% compared with the expected cropland yield. The rectangular thresholding method could be a more straightforward tool regarding computational cost since, e.g., the demarcation of rectangular areas is easier in any cultivated area, facilitating the implementation of spatial support plans for farmers. The proposal is to use the rectangular thresholding method as a planning tool, as the other methods may be used for more accurate estimations.

Keywords: Thresholding, Mahalanobis discriminant classifier, color detection, Normalized Difference Vegetation Index (NDVI), Image Processing, Precision Agriculture, Computer Vision, pixel-based segmentation, region-based segmentation.

[1] Universidad Pontifica Bolivariana, Seccional Montería. Departamento de Ingeniería electrónica. Montería, Colombia.
E-mail: david.hernandezmo@upb.edu.co; julio.gulfog@upb.edu.co; anam.lopezl@upb.edu.co
[2] Universidad del Sinú Elías Bechara Zainum. Departamento de Ingeniería Eléctrica. Montería, Colombia.
E-mail: claudiaserpa@unisinu.edu.co
* Autor de correspondencia: claudiaserpa@unisinu.edu.co

RESUMEN

Este estudio de investigación muestra una comparación entre el rendimiento de diferentes algoritmos de procesamiento de imágenes digitales basados en métodos de segmentación de imagenes para procesar imágenes multiespectrales satelitales de Índice de Vegetación de Diferencia Normalizada NDVI con el fin de estimar el rendimiento de las tierras de cultivo como una propuesta para apoyar las decisiones de planificación espacial. Las imágenes multiespectrales NDVI fueron capturadas del satélite Sentinel-2 L2-A con características distintivas, para ser procesadas a través de estos algoritmos implementados en una interfaz de software propia y amigable desarrollada en MATLAB App Designer. Estos se basan en la detección del color de la imagen, utilizando tres técnicas: el método de umbralización rectangular, el método de umbralización simple y la segmentación mediante el clasificador discriminante Mahalanobis. Las imágenes segmentadas se utilizaron para calcular el rendimiento de las tierras de cultivo en función de las variaciones del NDVI y las características de cada imagen analizada mediante un modelo lineal que asignaba un rendimiento a cada área segmentada en función de rangos de valores NDVI. La precisión del algoritmo se determinó en función del rendimiento esperado de las tierras de cultivo. Los resultados muestran que el método de umbralización rectangular tiende a promediar el valor del rendimiento de las tierras de cultivo en imágenes poco uniformes. En cambio, los métodos de umbralización por píxel y Mahalanobis mostraron un mejor rendimiento en imágenes NDVI muy poco uniformes, con desviaciones inferiores al 8% en comparación con el rendimiento esperado. Se puede concluir que el método de umbralización rectangular podría ser una herramienta de menor costo computacional, aunado al hecho de que la demarcación de áreas rectangulares es más fácil para delimitar e identificar para un agricultor en cualquier área cultivada real, facilitando la implementación de planes de apoyo para planificación espacial. Se propone utilizar el método de umbralización rectangular como herramienta de planificación, mientras los demás métodos pueden utilizarse para realizar estimaciones más precisas.

Palabras clave*: Umbralización, clasificador discriminante Mahalanobis, detección de color, índice de vegetación de diferencia normalizada (NDVI), procesamiento de imágenes, agricultura de precisión, visión por ordenador, segmentación basada en píxeles, segmentación basada en regiones.*

INTRODUCTION

Image processing and computer vision techniques have an enormous potential to be used as tools to estimate agricultural cropland yield based on temporal and spatial tracking, contributing to improving Precision Agriculture Systems [1] via crop data analysis, e.g., vegetation indexes coming from images of multispectral cameras. These data can correlate with agriculture variables like biomass, crop yield, plant leaf, fruit diseases, soil state, and others [2]-[4]. Image processing and computer vision techniques have been improved because they can be considered a starting point to estimate, predict, and later assess agriculture variables through feature extraction, as shown in this research study. The spectral and spatial analysis for classifying patterns contained on multispectral images of normalized difference vegetation index (NDVI) here correlates with cropland yield. An adequate assessment through classification and correlation will allow further implementation of intelligent technologies based on recent artificial intelligence techniques such as machine learning, deep learning, and semantic segmentation applied to agriculture [6].

Consequently, the performance of Precision Agriculture Systems could be strengthened through adaption to various complex situations, supporting management decisions according to the estimated variability of agriculture variables and needs of farmers. This approach is in agreement with the current definition of Precision Agriculture: "a management strategy that collects, processes, and analyzes temporal, spatial and individual data and combines them with other information to support management decisions according to the estimated variability, and thus improves efficiency in the use of resources, productivity, quality, profitability and sustainability of agricultural production"; therefore, an adequate correlation among data, assessment, and needs of farmers would increase success of end-users and other stakeholders to ensure effective co-production [6]. Regarding management decisions from analysis and monitoring of processed NDVI images, methods to systematize and improve efficiency in using NDVI image-based resources seeking to increase agricultural productivity have been recently tackled [3], [5], [7], [8]-[14]. NDVI images can correlate spectral and spatial digital information with the practical interest variables of usefulness in Precision Agriculture.

This paper compares the performance of different algorithms based on image processing segmentation-based computer vision techniques applied to Normalized Difference Vegetation Index (NDVI) multispectral images considered a trace of plant health [2]-[4]. NDVI images can be collected by satellites, uncrewed aerial vehicles (UAV), or drones [5] [6]. In this work, multispectral images were collected from Sentinel-2 L2-A satellite and segmented using an owned and friendly software interface developed in MATLAB App Designer to estimate agricultural cropland yield as a proposal for supporting such as determining irrigation areas, differential application of agrochemicals, crop status monitoring, water resource management decision; thus, promoting the planning and management of agricultural resources through monitoring crop growth assessment, disease prevention, quality tests, and harvest automation.

Segmented NDVI images were used to compute cropland yield depending on NDVI variations by intervals through a linear model that allocated a throughput to each segmented area depending on a specific NDVI range. Algorithms are based on image color detection, using three techniques: rectangular thresholding method that is a region-based segmentation, simple thresholding method that is a pixel-based segmentation, and segmentation through the method based on Mahalanobis discriminant classifier [11] [15]. The main challenge is that this proposed spectral and spatial method can be adapted, for instance, using other multispectral or hyperspectral images from cameras mounted on drones or any UAV. Similarly, these data sets can be easily interpreted for this method to be used by small and large farms as a decision-making tool to solve farmers' needs. Finally, algorithm accuracy was determined as a function of the expected cropland yield proposed as a linear mathematical model. Results show that the rectangular thresholding method tends to average cropland yield value in slightly non-uniform images. In contrast, thresholding by pixel and Mahalanobis methods performed better on highly non-uniform NDVI images. These last are more accurate with deviations less than 8% compared with expected cropland yield; however, the first one: the rectangular thresholding method, could be a better, more precise tool since e.g., the boundary of rectangles is easier for farmers in any cultivated area, facilitating to implement spatial support plans targeted to maximize cropland yield in an actual crop. The proposal is to use the rectangular thresholding method as a planning tool and the others as methods for a more accurate estimation.

MATERIALS AND METHODS

Owned software: Interfaces and operation

Figure 1 shows the software interfaces implemented in MATLAB App Designer to efficiently run up all the processes, step by step, to obtain any analysis and results described in this work. This software is called "DRIoT: Multispectral Image Processing for Precision Agriculture" (see Supplementary Material) and is an unpublished version of an image processing software targeted to support Precision Agriculture Systems. This software possesses an interface to upload images (Figure 1a), to later run algorithms such as the rectangular thresholding method, which is a region-based segmentation, simple thresholding method that is a pixel-based segmentation, and segmentation through the process based on Mahalanobis discriminant classifier [11] - [15] (Figure 1b up to Figure 1d). Furthermore, DRIoT software has other interfaces to run algorithms based on the "k- means" clustering method and Hough transform to detect shapes, borders, lines, and transition zone.

First, Figure 1a shows an interface to upload by the clicking of buttons: LOAD BAND 04 and LOAD BAND 08 at least two spectral images previously saved as .jpg or.png files in the computer that come from different sensors (an intensity detector at spectral range in a camera) that capture the reflectance (R) to a specific wavelength. The software works on two bands but can be extended to as many bands as possible. In this interface, the images were collected from the Sentinel-2 L2-A satellite that operates in

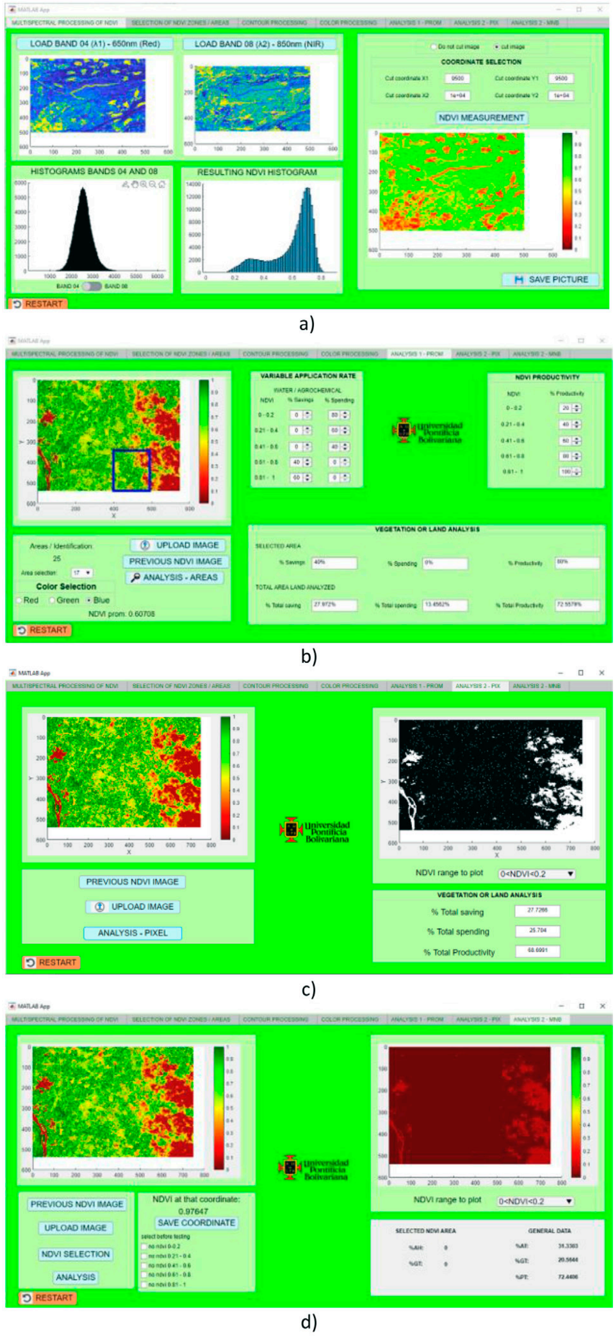

Figure 1. The DRIoT software interfaces developed in MATLAB App Designer use separate interfaces to run image processing algorithms. a) Interface to load images. b) Interface to process images using a simple thresholding segmentation method. c) Interface to process images using the rectangular thresholding segmentation method d) Interface to process images method based on Mahalanobis discriminant classifier.

many bands centered at determined wavelengths. The images used here were collected at two optical bands: Band 04 centered at 650 nm (red) and Band 08 at 850 nm (near-infrared well-known as NIR), as shown on the left side of Figure 1a. As from these two images, the NDVI multispectral image was obtained for applying pixel by pixel in each image band the equation (1), using the reflectance values at NIR (R_{NIR}) and Red (R_{red}):

$$NDVI = \frac{R_{NIR} - R_{red}}{R_{NIR} + R_{red}} \tag{1}$$

as shown on the right side of Figure 1a after clicking on NDVI MEASUREMENT button. This interface was provided with other functionalities, such as cutting images and histograms, to analyze image uniformity. Figure 1a shows images in each band (two bands) of 500 x 500 pixels and NDVI image of 500 x 500 pixels. Images were previously processed from bigger images through the cutting algorithm to reduce X and Y sizes.

Algorithm segmentation and yield linear model proposed

Once the NDVI (dimensionless value) multispectral image is obtained, this is processed using the three algorithms mentioned above (See mathematical descriptions and their algorithms in Appendix A). For this, Figure 1b shows a second interface to process the NDVI multispectral images using the rectangular thresholding segmentation method, in which segmented areas are denoted by a square (see the blue square in Figure 1b. Similarly, Figure 1c shows a third interface to process NDVI multispectral images using the simple thresholding segmentation method. Finally, Figure 1d shows an interface to process NDVI multispectral images based on the method of the Mahalanobis discriminant classifier (NDVI segmented images in Appendix B).

Interfaces shown in Figure 1b to Figure 1d are provided with a panel to show estimated cropland yield once the NDVI multispectral image is processed using the segmentation algorithms. A spectral and spatial linear model relating NDVI and spatial distribution of cropland yield is proposed in this research work to estimate yield over the whole NDVI image [3]. To estimate the yield in the entire NDVI image, a spectral and spatial linear model that relates the NDVI and the spatial distribution of cropland yield are proposed in this research work [3]. In this, NDVI variations on each processed image through the segmentation algorithms are modeled through a linear equation that allocates a throughput to each segmented area depending on a specific NDVI range or interval. As the NDVI values are related to plant health, and therefore implicitly with the yield; then, NVDI values near or less than 0.5 would represent a low yield, and it will be a high yield if NDVI is approaching 1. Based on this hypothesis, each segmented area yield will be assigned using these yield values or "weight" as shown in Table 1:

Table 1. Yield assigned depending on NDVI interval detected on each segmented area.

NDVI interval	Yield assigned (%)
0-0.2	20%
0.21-0.4	40%
0.41-0.6	60%
0.61-0.8	80%
0.81-1	100%

Linear equation can be expressed as:

$$\frac{\%Y = NDVI(i,j)_{(0-0.2)}(20) + NDVI(i,j)_{(0.21-0.4)} (40) + NDVI(i,j)_{(0.41-0.6)}(60) + NDVI(i,j)_{(0.61-0.8)} (80) + NDVI(i,j)_{(0.81-1)}(100)}{NDVI(i,:) * NDVI(:,j)} \tag{2}$$

where NDVI(i, j) (0-0.2) represents the segmented pixels on the image with NDVI values between 0 to 0.2, NDVI (i, j) (0.21-0.4) represents the segmented pixels on the same image with NDVI values between 0.21 to 0.4, NDVI (i, j) (0.41-0.6) represents the segmented pixels on the image with NDVI values between 0.41 to 0.6, NDVI (i, j) (0.61-0.8) represents the segmented pixels on the image with NDVI values between 0.61 to 0.8, and NDVI (i, j) (0.81-1) represents the segmented pixels on the image with NDVI values between 0.81 to 1. NDVI (i,:) and NDVI (:,j) are the NDVI values in rows and columns where NDVI (i,:)*NDVI (:,j) is the area of the entire image in terms of pixel numbers. 20, 40, 60, 80, and 100 are the linear coefficients that represent the yield assigned to each segmented pixel or area. %Y is the yield estimated in the entire image, computed using this weighted sum of equation (2) applied after segmentation. In this proposed model to estimate cropland yield in the entire NDVI image, the yield assigned (see the second column in Table 1) can be modified according to spatial planning decisions and desired or expected outcomes, depending on other environments or external conditions. Monitoring yield values, for instance, based on a time series or other spectral images at different wavelengths, can be used for decision support such as determining irrigation areas, differential application of agrochemicals, crop status monitoring, water resource management decision, disease prevention, quality tests, and harvest automation.

RESULTS

Productivity estimation using slightly non-uniform images

As the first step, a performance test of the linear model was made. For this, six NDVI multispectral images (Figure 2) were chosen with specific features to test the performance of segmentation algorithms and the linear yield models. These NDVI images are classified as highly uniform by analyzing their histogram. In an ideal framework, images with all red (NDVI around 0.2), yellow (NDVI approximately 0.6), or green (NDVI around 1) content should come in yield expected values at about 20%, 60%, or 100%, according to Table 1 and color bars from Figure 1a, after running any of three segmentation algorithms. The main goal of using these slightly uniform images with slight non-uniformity is to test the sensitivity to minor variations of NDVI on the image, which can be observed as pixels with a slight deviation from the desired principal value either, red, yellow, or green. Thus, these images are considered mainly red, yellow, or green, but only partially.

Yield computed in the entire NDVI images is shown in Table 2.

By using these kinds of images with quasi-uniform NDVI distribution, the rectangular thresholding

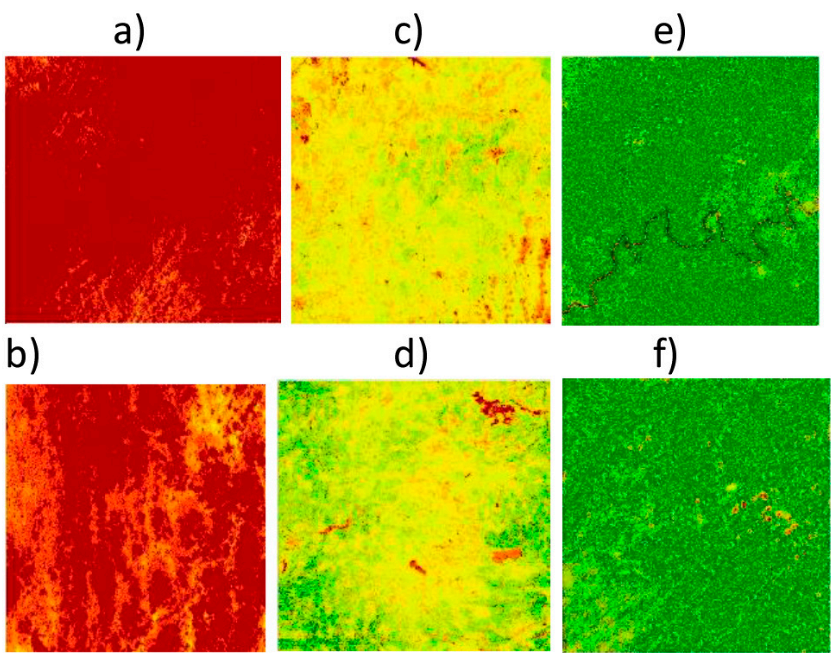

Figure 2. NDVI Images to test the yield linear model performance and its dependency with NDVI distribution in slightly non-uniform. a) Image with a resolution of 50 m/pixel with high red content. b) Image with a 40 m/pixel resolution with high red content. c) Image with a resolution of 40 m/pixel with high yellow content. d) Image with a 40 m/pixel resolution with high yellow content. e) Image with a 40 m/pixel resolution with high green content. f) Image with a 40 m/pixel resolution with high green content.

Table 2. Values of estimated yield using the proposed linear model, the three segmentation methods by using slightly non-uniform images of Figure 2.

Figure/Yield (%)	Rectangular thresholding	Simple thresholding	Mahalanobis
Figure 2a	20	21.4	22
Figure 2b	20.20	27.75	28.11
Figure 2c	60	59.25	66.49
Figure 2d	60.71	62.21	63.61
Figure 2e	99.9	91.95	97.95
Figure 2f	99.51	90.18	98.7

segmentation method (region-based segmentation) averages NDVI values that approximate the main pixel's value of the image with NDVI value corresponding to red (0.2), yellow (0.6), and green (1), and its previously assigned yield value as red (20%), yellow (60%) and green (100%). In contrast, the simple thresholding segmentation method (pixel-based segmentation) and the method of Mahalanobis discriminant classifier compute productivity values slightly near the average but with deviations of up to a maximum of 10% compared with expected values of red (20%), yellow (60%), and green (100%), in this case, is evident the effect of introducing non-uniformity in the images in the computing of yields. The single threshold segmentation method and the Mahalanobis discriminant classifier method were found to be very sensitive to non-uniformity in the NDVI images to calculate the yield values more accurately.

Productivity estimation using highly non-uniform images

As the second step, a linear model performance test was made. In this case, three NDVI multispectral images (Figure 3) were chosen to test the performance of segmentation algorithms and the linear yield models. The images are highly non-uniform to apply a segmentation algorithm and estimate the percentage productivity.

Yields computed for the entire image are shown in Table 3. As shown in the previous section, the rectangular thresholding method tends to average the NDVI values approximating the main pixel's value of the image with NDVI value corresponding to green (yield of 80%) and yellow (yield of 60%). Regarding these values (80% for Image 7,80% for Image 8, and 60% for Image 9), the maximum error is approximately 4% compared with the expected main pixel's value. The simple thresholding and Mahalanobis methods exhibit differences with minor deviations of up to 8% concerning values obtained by the rectangular thresholding method. In this case, the effect of introducing highly non-uniformity on the images is more evident. Therefore, as previously discussed, the simple thresholding and the Mahalanobis discriminant classifier are

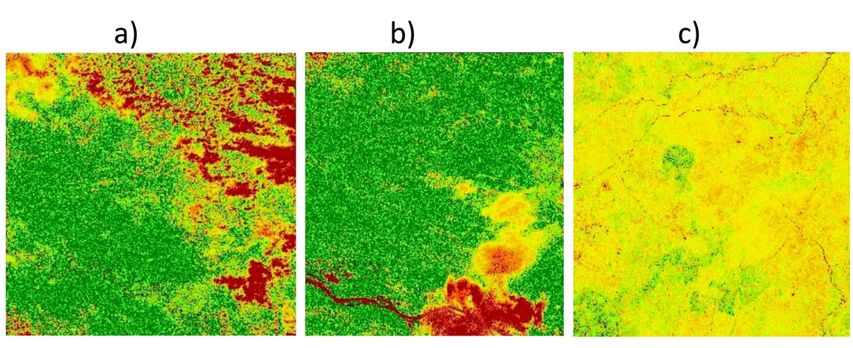

Figure 3. NDVI Images to test the performance productivity of the linear model and its dependency with NDVI distribution in highly non-uniform images a) Image with a 50m/pixel resolution. b) Image with a 40m/pixel resolution. c) Image with a 40m/pixel resolution.

Table 3. Values of estimated yield using the proposed linear model, the three segmentation methods using non-uniform images from Figure 3.

Figure/Yield (%)	Rectangular thresholding	Simple thresholding	Mahalanobis
Figure 3a	79.79	71.10	71.32
Figure 3b	77.01	81.74	87.78
Figure 3c	60	60.28	64.51

more suitable for segmenting and calculating the yields in a complete image, showing more similar values and higher accuracy.

DISCUSSION

The results show how segmented images can be used to estimate cropland yield depending on NDVI value distribution on images by applying a linear model and assuming they represent the digital record of any crop biomass in the entire image. Linear coefficients and even other models can be adapted depending on planning decisions and prior knowledge of agronomists, local policies, desired or expected yield, or other variables involved in cropland planning [25]. The method implemented in this software is expected to be employed to analyze the crop through the data obtained using the segmented image; it could also be easily interpreted and used by small and large farms as a decision-making instrument to solve their needs. Future research can be addressed to use the potentiality to complete information of the time, spatial and spectral information of other kinds of visible RGB, multispectral hyperspectral images to distinguish, for instance, soil, vegetation, fruit, crops, and other features; based on data for further processing towards the retrieval of conclusive information.

Several other segmentation techniques, such as semantic segmentation, other classification techniques such as k-means, or any other machine learning and deep learning method, can be adapted to analyze images as a digital representation of crops involving all variables [26-28].Considering that these images represent crop biomass, these results show that the rectangular thresholding method tends to average cropland yield value in slightly non-uniform images. In contrast, thresholding by the pixel and Mahalanobis discriminant classifier methods showed better performance on highly non-uniform NDVI images. These last are more accurate with deviations less than 8% compared with expected cropland yield. Nonetheless, the first one: the rectangular thresholding method, could be a simpler tool to use in the planning process since, e.g., the differentiation of rectangles is easier in any cultivated area, facilitating to use these results to be implemented as spatial support plans targeted to maximize cropland yield in an actual crop. Following these results, the proposal is that these methods should be combined as a decision-making tool, where the rectangular thresholding method could be used as a planning tool, and the others as methods for a more accurate estimation of the cropland yield.

CONCLUSIONS

This research study presented a method for estimating yield using a linear model depending on the NVDI and its distribution on the image detected by segmentation algorithms. Three algorithms based on rectangular thresholding, simple thresholding, and the Mahalanobis discriminant classifier methods are implemented using a friendly and own software developed in MATLAB App Designer. The results indicate that these segmentation methods can be used to estimate farmland yields to implement a precision farming system consistent with farmers' needs. The simple thresholding and Mahalanobis methods are more accurate; however, thresholding by areas can be simpler to implement as a tool for planning decisions. This method can be extended with other multispectral images independent of the resolution.

ACKNOWLEDGEMENT

This study was financially supported with resources from Universidad Pontificia Bolivariana Seccional Montería under Innova Project Nº 228-07/18-G007.

REFERENCES

[1] M. Roberto do Santos, G.A. Madalozzo, J. M. Fernandes, and R. Rieder, "Fenômica: A Computer Vision System for High-Throughput Phenotyping," *International Journal of Agricultural and Environmental Information Systems*, 2020, vol. 11, no. 1. pp. 1-22, doi:10.4018/IJAEIS.2020010101.

[2] V. Singh, A.K. 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, doi: 10.1016/j.inpa.2016.10.005.

[3] H. Tian, T. Wang, Y. Liu, X. Qiao, and Y. Li, "Computer vision technology in agricultural automation -A review," *Information Processing in Agriculture*, vol. 7, no. 1, 2020, pp. 1-19, doi: 10.1016/j.inpa.2019.09.006.

[4] S. Khanal, K. KC, J.P. Fulton, S. Shearer, and E. Ozkan, Remote Sensing in Agriculture-Accomplishments, Limitations, and Opportunities, *Remote Sens*., vol. 12, no. 22, p. 3783, 2020, doi:10.3390/rs12223783.

[5] T. Talaviya, D. Shah, N-Patel, Hiteshri Yagnik, Manan Shah, "Implementation of artificial intelligence in agriculture for optimization of irrigation and application of pesticides and herbicides," *Artificial Intelligence in Agriculture*, vol. 4, pp. 58-73, 2020, doi: 10.1016/j.aiia.2020.04.002.

[6] D. Ma, L. Wang, L. Zhang, Z. Song, T.U. Rehman, and J. Jin, "Stress Distribution Analysis on Hyperspectral Corn Leaf Images for Improved Phenotyping Quality," *Sensors*, vol. 20, no. 13, p. 3659, 2020, doi: 10.3390/s20133659.

[7] M.S. Boori, K. Choudhary, and A.V. Kupriyanov, "Crop growth monitoring through Sentinel and Landsat data based NDVI time-series," *Computer Optics*, 2020, vol. 44, pp. 409-419, doi: 10.18287/2412-6179-CO-635.

[8] R.J. Haggar, C.J. Stent, and J. Rose, "Measuring spectral differences in vegetation canopies by a reflectance ratio meter," *Weed Research*, 1984, vol. 24, no. 1, pp. 59-65, doi: 10.1111/j.1365-3180.1984.tb00571.x.

[9] R. Bhattarai, P. Rahimzadeh-bajgiran, A. Weiskittel, D.A. Maclean, "Sentinel-2 based prediction of spruce budworm defoliation using red-edge spectral vegetation indices," *Remote Sensing Letters,* vol. 11, no. 8, pp. 777-786, 2020, doi:10.1080/2150704X.2020.1767824.

[10] D. Luebeck *et al.*, "Drone-borne Differential SAR Interferometry," *Remote Sens.* vol. 12, no. 5, p. 778, 2020, doi:10.3390/rs12050778.

[11] L. Wang *et al.*, "Precise Estimation of NDVI with a Simple NIR Sensitive RGB Camera and Machine Learning Methods for Corn Plant," *Sensors*, vol. 20, no. 11, p. 3280, 2020, doi: 10.3390/s20113208.

[12] H. Li, *et al.*, "Monitoring Wheat Growth Using a Portable Three-Band Instrument for Crop Growth Monitoring and Diagnosis," *Sensors*, vol. 20, no. 10, p. 2894, 2020, doi:10.3390/s20102894.

[13] A. Camps, A. Alonso-Arroyo, H. Park, R. Onrubia, D. Pascual, and J. Querol, "L-Band Vegetation Optical Depth Estimation Using Transmitted GNSS Signals: Application to GNSS-Reflectometry and Positioning," *Remote Sensing*, vol. 12, no. 15, p. 2352, 2020, doi: 10.3390/rs12152352.

[14] L. Cui, Z. Wang and L. Deng, "Vegetation dynamics based on NDVI in Yangtze River Basin (China) during 1982-2015," *IOP Conference Series: Materials Science and Engineering*, vol. 780, no. 6, p. 062049, 2020, doi:10.1088/1757-899X/780/6/062049.

[15] W. Li *et al.*, "A hybrid modelling approach to understanding adoption of precision agriculture technologies in Chinese cropping systems," *Computers and Electronics in Agriculture*, vol. 172, p. 105305, 2020, doi: 10.1016/j.compag.2020.105305.

APPENDIX A

Mathematical basis of segmentation methods

Segmentation refers to the process of partitioning a digital image into multiple segments with similar features by assigning a "label" to every pixel so that pixels with the same label share particular visual common characteristics. Image segmentation results in a set of labeled segments that collectively shape the entire image. Each pixel in a region is similar concerning some characteristic or computed property, such as color, intensity, or even texture. One of the ways to recognize interest objects is made by segmentation methods to find suitable local features that can be distinguished from other objects and the background, checking each pixel to see whether it belongs to an object of interest or not. Segmentation methods are used to characterize the health of cultivated areas through NDVI values. This Appendix shows the mathematical basis of segmentation methods used here: pixel-based segmentation (A1), Region-based segmentation (A2), and Mahalanobis discriminant classifier (A3).

A1. Pixel-based segmentation:

Pixel-based segmentation is the simplest approach used for segmentation. This method separates the image pixels according to a threshold value into two categories defined by a specific value. Pixel-based segmentation method can be mathematically modeled by a thresholding function *T* defined as:

$$T = T\left[x, y, p\left(x,y\right), f\left(x,y\right)\right] \qquad \text{(A1)}$$

where *x* and *y* are the coordinates of the pixel in image *f*, *f(x,y)* is the pixel feature (color, intensity, or texture) at these coordinates, *p(x,y)* represents some local property of the point for discrimination. From these parameters, a function *g (x, y)* is defined by the equation:

$$g\left(x,y\right) = \left\{ \begin{array}{c} 0 \; si \, f\left(x,y\right) > T \\ 1 \; si \, f\left(x,y\right) \leq T \end{array} \right\} \qquad \text{(A2)}$$

and, it is assigned 0 value if pixels correspond to the desired object and 1 otherwise. This mathematical model was implemented in MATLAB to analyze the images by this method. The image is transformed into black and white, then the desired threshold values are defined to find the segmented areas. A part of the code can be observed here:

```
limiteinf = app.Limiteinf.Value;
limitesup = app.Limitesup.Value;
    I4 = imbinarize (I, limitesup);
    I5 = imbinarize (I, limiteinf);
    I6 = not (I4);
    I7 = and (I6, I5);
```

where I4 to I7 are the thresholds values of segment the images.

A2. Region-based segmentation

This method focuses on an essential aspect of the segmentation process where there could be missed pixels belonging to an object that cannot be detected with pixel-based techniques. Thus, if the image is highly non-uniform, many pixels can be classified as an object pixel judging solely on their gray or color value, independently from the context, i.e., it cannot be identified as a part of an object. Isolated points or small areas could be classified as object pixels. The features do not represent isolated pixels but a small neighborhood belonging to an object. This problem can be solved by iteratively using a procedure in which feature computation and segmentation are alternately performed using interval values or means values that characterize the desired object considering the neighborhood of the object to limit the object edges depending on the location of the center pixel. For better results, feature analysis and segmentation can be repeated until the procedure converges to a stable or desired outcome based on the previously assigned mean value. These concepts described here were implemented in MATLAB. The image is transformed into black and white, defined to find the segmented areas through a thresholding value to compute a mean of the segmented image. Part of the code can be observed here:

```
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;
```

A3. Mahalanobis discriminant classifier

Covariance is a statistical measurement that indicates how strong the correlation between 2 or more random variables is. For a set of 2 or more random variables, a covariance matrix is used and is defined as:

$$V_{ij} = \mathrm{cov}\left(x_i, x_j\right) = \sum_{k=1}^{N} \frac{\left(x_{i_k} - \bar{x}_i\right)\left(x_{j_k} - \bar{x}_j\right)}{N} \quad \text{(A3)}$$

V_{ij} is the matrix input *(i, j)*. For an RGB image, a set of three random variables are built (R, G, and B), the covariance matrix will be the following relationships:

Table A1. Covariance matrix relationships for an RGB image.

i/j	R	G	B
R	*cov (R, R)*	*cov (R, G)*	*cov (R, B)*
G	*cov (G, R)*	*cov (G, G)*	*cov (G, B)*
B	*cov (B, R)*	*cov (B, G)*	*cov (B, B)*

The Mahalanobis discriminant classifier is based on correlations among variables with different patterns that can be identified and analyzed. It allows calculating the distance between a weighted reference color and its variance of each image component array (i.e., RGB or CMYK for commonly used images). It is an applicable method to determine the similarity of an unknown set of samples to a known one. It differs from the Euclidean distance in that it considers the correlations between the data set:

$$D = \sqrt{\left(x - y\right)^T C^{-1}\left(x - y\right)} \quad \text{(A4)}$$

C^{-1} is the non-singular covariance matrix and $(x–y)^T$ is the transpose of the data; the Malahanobis distance is equal to Euclidean distance if the matrix C is equal to an identity matrix. In this distance, the image coordinates with the desired color value are selected to perform the color segmentation, thus taking the three RGB color components. Then the mathematical formula is applied, creating a new matrix with the result of the segmented image. A part of the implemented code in MATLAB is:

```
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
```

APPENDIX B

NDVI Segmented images

Some examples of NDVI segmented images of Figure 2 and Figure 3 used in this work are shown in Figure B1. Figure B1 from a) to e) (a, b, c, d, and e) shows the segmented images of Figure 3b using the pixel-based segmentation method. Each part of the segmented image is represented as a black-and-white image according to the NDVI interval values. Figure B1a is the segmented image with NDVI values in the interval from 0 to 0.2 depicted as white pixels, and black pixels are the background or the part of the images with values that are not inside of the NDVI interval. Figure B1b is the segmented image with NDVI values from 0,21 to 0.4 in the interval. Figure B1c is the segmented image with NDVI values from 0.41 to 0.6 the interval. Figure B1d is the segmented image with NDVI values from 0.61 to 0.8 in the interval. Figure B1e is the segmented image with NDVI values in the interval from 0.81 to 1. Figure B1 from f) to j) (f, g, h, i, and j) shows five of all segmented images (63 in total) of Figure 2b using the region-based segmentation method. Each part of the segmented image is depicted in the original color NDVI scale. With this method, each segmented region is enclosed by a square with the NDVI mean value of the segmented region. Figure B1 from k) to

o) (k, l, m, n, and o) shows the segmented images of Figure 3a using the Mahalanobis segmentation method. Each part of the segmented image is depicted as a color image with a vino background that does not correspond to the NDVI interval values. Figure B1k is the segmented image with NDVI values in the interval from 0 to 0.2, and pixels different from the background are part of the segmented image depicted in the original color NDVI scale. Figure B1l is the segmented image with NDVI values from 0,21 to 0.4 in the interval. Figure B1m is the segmented image with NDVI values from 0.41 to 0.6 in the interval. Figure B1n is the segmented image with NDVI values from 0.61 to 0.8 in the interval. Figure B1o is the segmented image with NDVI values from 0.81 to 1 in the interval.

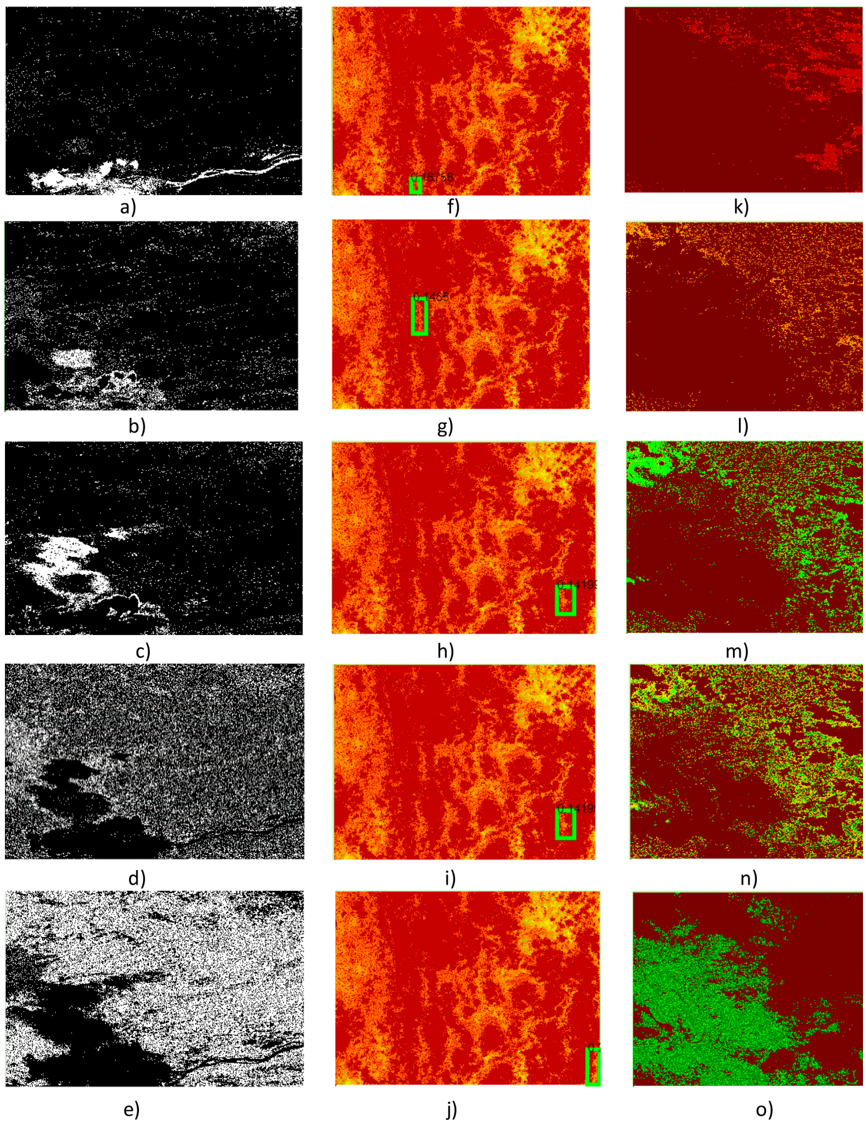

Figure B1. Some examples of NDVI segmented images. Figures from a) to e) are the 5 five segmented images using the pixel-based segmentation method. a) NDVI from 0 to 0.2, b) NDVI from 0.21 to 0.4, c) NDVI from 0.41 to 0.6, d) NDVI from 0.61 to 0.8, e) NDVI from 0.8 to 1. Figures from f) to j) are the five 5 of all 63 segmented images corresponding to NDVI values of the images. Figures k) to o) are the 5 five segmented images by the Mahalanobis segmentation method. k) NDVI from 0 to 0.2, l) NDVI from 0.21 to 0.4, m) NDVI from 0.41 to 0.6, n) NDVI from 0.61 to 0.8, o) NDVI from 0.8 to 1. Other segmented images are not presented here by simplicity, and they are used to compute the proposed linear productivity model.