Ecological Informatics 66 (2021) 101465

Contents lists available at ScienceDirect

Ecological Informatics

journal homepage: www.elsevier.com/locate/ecolinf

Machine learning and SLIC for Tree Canopies segmentation in urban areas

José Augusto Correa Martins [a,*], Geazy Menezes [b], Wesley Gonçalves [a,b], Diego André Sant'Ana [c], Lucas Prado Osco [a], Veraldo Liesenberg [d], Jonathan Li [e], Lingfei Ma [e], Paulo Tarso Oliveira [a], Gilberto Astolfi [b,c], Hemerson Pistori [c], José Marcato Junior [a]

[a] Faculty of Engineering, Architecture and Urbanism and Geography, Federal University of Mato Grosso do Sul (UFMS), Campo Grande, Mato Grosso do Sul, Brazil
[b] Faculty of Computer Science, Federal University of Mato Grosso do Sul (UFMS), Campo Grande, Mato Grosso do Sul, Brazil
[c] Environmental Science and Sustainability, INOVISÃO Universidade Católica Dom Bosco (UCDB), Campo Grande, Mato Grosso do Sul, Brazil
[d] Department of Forest Engineering, Santa Catarina State University (UDESC), Lages, Santa Catarina, Brazil
[e] Department of Geography and Environmental Management and Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada

ARTICLE INFO

Keywords:
Urban environment
Machine learning
Remote sensing
Photogrammetry
Computer vision

ABSTRACT

This paper presents a novel approach combining the Simple Linear Iterative Clustering (SLIC) superpixel algorithm with a Convolutional Neural Network (CNN) over high-resolution imagery to detect trees in a typical urban environment of the Brazilian Cerrado biome. Our analysis approach for better results uses the deep learning classifier ResNet-50, with a variation in the batch size and five traditional shallow learning methods. The results were processed and avaliated using mainly accuracy as a metric, but we show that the accuracy poorly represent the overlap between the manual annotation and the resulting map, so we bring the IoU metric results to better show the Network learning classification maps results. The combined SLIC algorithm and the best CNN resulted in an accuracy of 93.20%, IoU of 0.700 and a variation of 1% for difference in the area of tree canopies if compared to our labels, while the best shallow presented an accuracy of 91.70%, IoU of 0.200 and a variation in area of 12.52%. Demonstrating that the proposed CNN method is suitable for segmenting trees from high-resolution images acquired over urban environments. The segmentation with SLIC and CNN can provide very useful results for urban management using low cost RGB images. Such outcomes are of great interest for local managers since reliable maps showing the spatial distribution of trees in urban areas are often required for many applications.

1. Introduction

Forests harbor a significant part of the planet's biodiversity, promote the water cycle, recycles carbon from the atmosphere, among many other beneficial environmental services (Kwong and Fung, 2020). The efforts in mitigating global forest cover and biodiversity loss often are focused on preserving native natural habitats, but that is often not the case with urban forests or groves, despite the essential services that urban vegetation and biodiversity perform to the city such as, mitigating heat islands, Mitigating storm-water runoff and flooding, promoting the ecological balance, improving the resident's quality of life, barrier for pollutants, and many others (Baró et al., 2014; Shojanoori and Shafri, 2016). Managers require reliable, accurate, and up-to-date information to maintain a sustainable environment, like vegetation health and coverage area. These characteristics contribute to understand the urban forest dynamics better and implement efficient decision-making strategies related to its management and sustainability (Fassnacht et al., 2016).

As field measurements for tree counting are time-consuming and may be expensive, remote sensing systems acquiring high spatial resolution images and coupled with machine learning techniques may overcome this limitation (Shafri et al., 2011). However, this task becomes more challenging for urban areas because of its complexity, which comes in the form of the intrinsic features of the urban landscape that present objects with a wide variety in shapes, colors, and textures. Also, the structure of the area and small tree crown can result in mixed pixels (Shojanoori and Shafri, 2016). Shojanoori and Shafri (2016) recommended that further studies targeting urban forests shall focus on

* Corresponding author.
E-mail addresses: jose.a@ufms.br (J.A. Correa Martins), geazy.menezes@ufms.br (G. Menezes), wesley.goncalves@ufms.br (W. Gonçalves), diego.santana@ifms.edu.br (D.A. Sant'Ana), pradoosco@gmail.com (L.P. Osco), veraldo.liesenberg@udesc.br (V. Liesenberg), junli@uwaterloo.ca (J. Li), L53ma@uwaterloo.ca (L. Ma), paulo.t.oliveira@ufms.br (P.T. Oliveira), gilberto.astolfi@ifms.edu.br (G. Astolfi), pistori@ucdb.br (H. Pistori), jose.marcato@ufms.br (J.M. Junior).

https://doi.org/10.1016/j.ecoinf.2021.101465
Received 15 May 2021; Received in revised form 15 October 2021; Accepted 18 October 2021
Available online 29 October 2021

three main issues, being the first one on the development of vegetation indices. Second, to promote high-accuracy algorithms to distinguish urban tree species automatically via high-resolution multi-spectral imaging. Finally, to develop automatic tree counting methods.

For this task, we will combine the use of machine learning and remote sensing. Machine learning is the science (and art) of programming computers so they can learn from data (Geron and Géron, 2017). Inside the machine learning field, we have the Shallow learning algorithms (SL) and the current state-of-the-art method for image semantic segmentation, called deep learning (DL). With a simplistic view, the main difference between them is the number of learning layers; in the DL algorithms, we have many more learning layers than in the SL algorithms. Networks with more than tree layers are already considered to be deep, but some Deep Networks such as ResNet101 (He et al., 2013), have hundreds of layers. With the DL algorithms as described by Goodfellow et al. (2016), is possible to the machine to learn specific and adaptable spatial features directly from the images, creating a system that is capable of mapping a raw input into the desired output with little human interference, in the case of Unsupervised learning the machine tries to emulated a difference in the data and brings the results in the form of a map. Also, inside the machine learning field, we have the superpixel algorithms, which group pixels into perceptually meaningful atomic regions (Achanta et al., 2012), making it easier for the machine to process the data. CNN's are largely applied for image recognition and segmentation, and previous studies also considered the combination of superpixels and CNN for different image datasets and obtained notable results (dos Santos Ferreira et al., 2017; dos Santos et al., 2019; Zhao et al., 2017). Automatic learning algorithms are becoming an indispensable tool for remote sensing applications like object detection and segmentation (Wan et al., 2018; Shelhamer et al., 2017; Noh et al., 2015; Sothe et al., 2020; Ma et al., 2019; Piazza et al., 2016). In the forest context, we also have many examples of applications such as Feng et al. (Feng et al., 2015) that proposed a method based on the Random Forest algorithm and texture analysis to differentiate land covers of urban vegetated areas. Onishi and Ise (2018) proposed a deep learning segmentation method making a model for object identification with NVIDIA DIGITs. Heinrich (Heinrich, 2016) proposed an object identification method using a deep learning framework and GoogLeNet (Szegedy et al., 2015) as a transfer learning model. Furthermore, there are in-numerous methods to obtain valuable land cover information, including tree species detection (dos Santos et al., 2019), forest inventory (Puliti et al., 2015), forest-wildfires tracking (Tang and Shao, 2015), citrus tree counting (Osco et al., 2019).

All above mentioned tasks are solved with machine learning techniques. Machine learning is the science (and art) of programming computers so they can learn from data (Geron and Géron, 2017). Adhikari et al. (2021) proposed a Simple Linear Iterative Clustering (SLIC) segmentation and a classification with a Random Forest (RF) algorithm for segmenting tree canopies in an orchard area. The authors used images with a spatial resolution of 0.13m and achieved an overall accuracy of 93%. However, the proposed approach was not yet evaluated over complex environments in which tree canopies does not follow regular patterns. Additionally, the comparison of different ML algorithms has not been evaluated yet. Therefore, the combined use of SLIC and different ML are still necessary, especially over complex urban environments.

We combine high resolution imagery and machine learning methods to segment tree canopies in a typical urban environment accurately. The urban tree segmentation task refers to the automated classification of each pixel of a given image into a tree or background, and that is a challenging problem because urban forests are a particular case of vegetation with specific characteristics as an object they are diverse in shape, color, texture and dense or poor in numbers, blended with other metropolitan elements and grouped or remotely localized (Jensen et al., 2009). Therefore, automated urban forest mapping is a challenging computational task. With this objective in target, we want to propose a model for accurate and automatic segment groups and singular trees combining remote sensing and ML. This task has been an extended date objective of many researchers and land-use managers (Fassnacht et al., 2016). The combination of ML techniques with UAV-based imagery provides a robust and cost-efficient method for data extraction and analysis, assisting in environmental management strategies, climate change actions, and many others. Furthermore, the association of ML with UAV imagery makes possible a more refined (singular trees) and automatic tree detection with high accuracy under challenging environments.

The proposed approach analyzes one Convolutional Neural Network (CNN) performance and compares it to a set of five shallow learning algorithms. The input data was extracted from an orthomosaic using a SLIC superpixel technique. The SLIC presents an excellent boundary detection capabilities as presented in Achanta et al. (2012), which is fundamental while working with semantic segmentation tasks. This study's contribution is to present an automated and accurate method suitable for segmenting individual trees or groups of trees from a typical urban environment inside the Cerrado biome, with high-resolution RGB imagery obtained with consumer-grade UAV, and using open-source software. This way provides an accessible and accurate quantification of the vegetated cover inside any given area of a city, giving information support for future decision making, planning, and improving the environmental services provided by the urban trees.

2. Materials and methods

2.1. Workflow and hardware

The DL framework of this research, see Fig. 1, is presented in the format of a Convolutional Neural Network (CNN) with a supervised learning, we inputted the machine with our label and the raw image raster file, processed the data and evaluated the results. The methodology and results, and discussion are arranged in a logical manner. First, we acquire the images and pre-process the image file by creating the orthomosaic and rectified it. Secondly, we create manual labels of our target object (trees) using the orthomosaic. Thirdly, with the orthomosaic we created the superpixels and extracted the attributes that the classifiers will use to classify the image. Fourthly, we classified the orthomosaic with the selected algorithms. Lastly, we analyzed and discussed the results.

All the models were trained in a hardware with the following configurations: Processor – AMD Ryzen 7 (1800x); RAM memory – 2 × 16 GB DDR4 de 2400mhz; Hard-Drive: 240 GB SSD; Graphical Card – NVIDIA TITAN Xp, with specs of: Frame Buffer G5X; Memory Speed 11.4 Gbps; Boost Clock 1582 MHz, architecture NVIDIA Pascal™.

2.2. Data acquisition

The study was carried out in an area located at the Federal University of Mato Grosso do Sul (UFMS), in the municipality of Campo Grande, Mato Grosso do Sul state, Brazil (20°26′37″S, 54°38′52″W) presented in Fig. 2(a). The study area has non-uniform crown densities, with isolated trees and some groves. Aside from just trees, the UAV flight also registered streets, cars, buildings, and other objects. These characteristics were considered when choosing this sample for the demonstration of the proposed method, of detecting trees in urban environments.

The flight campaign was conducted using Pix4DCapture software, and more details can be observed in Table 1. The UAV system kit consists of: a sensor for data acquisition, autopilot for controlling the platform in a determined track, GPS for orientation and navigation, Inertial Measurement Unit (IMU) for attitude measurement, and a data link signal transfer, ground station, and control points geolocalization. More information about the sensor and flight are presented in Table 1.

The orthorectification process intends to rectify each image and group them into a single one with a constant scale. This is often a

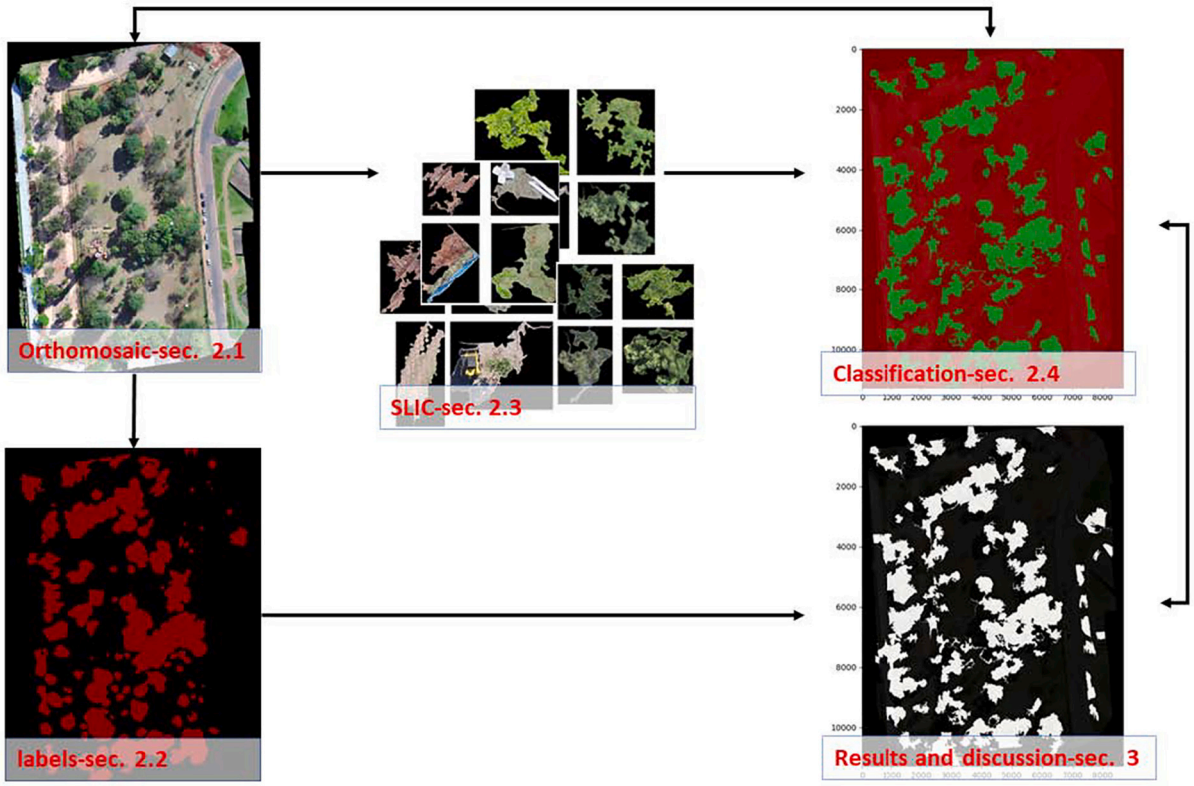

Fig. 1. Overview of the workflow and learning process. In this figure sections presents in the text are refereed as sec.

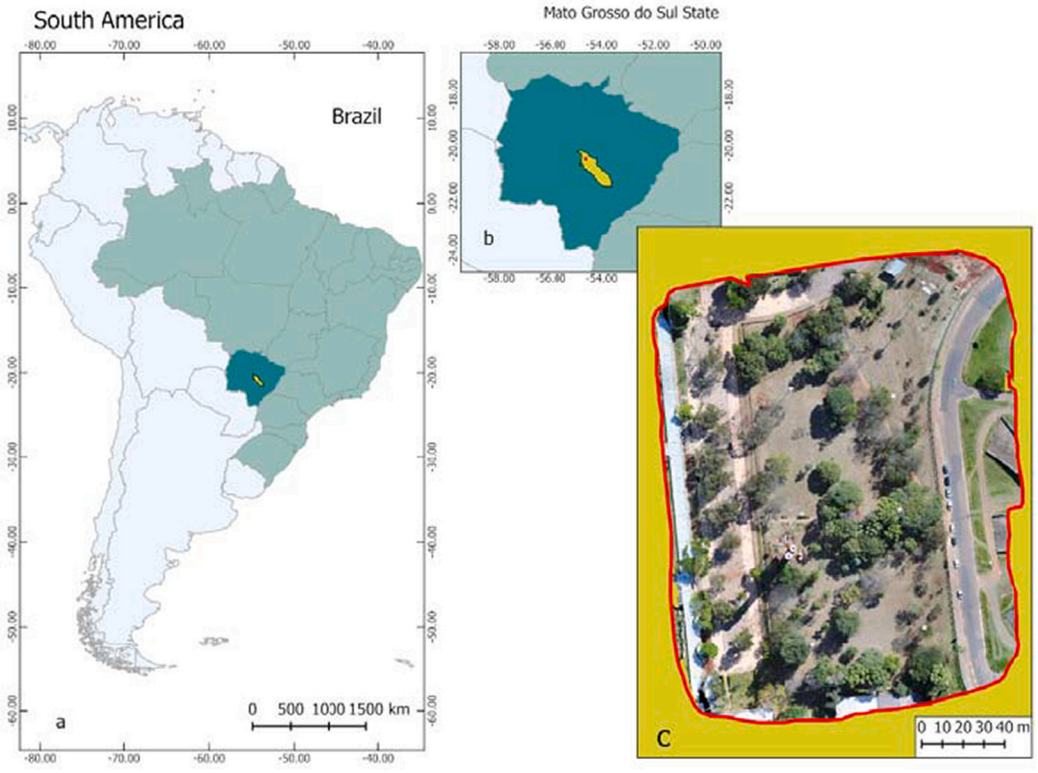

Fig. 2. (a) Continental location of the area, (b) regional location of the area, (c) study site where the flight mission was conducted, inside UFMS Campus with the UAV, on 26 May 2018.

Table 1
Weather condition and image capturing specifications.

Date (month/day/year)	05/26/2018
Local time (hh:mm)	10:00
Temperature (°C)	24.18
Dew point (%)	59.42
Wind direction	N
Wind Speed max (km/h)	14.04
Weather condition	Fair
Aircraft	Phantom 4 Advanced
Sensor	1″ CMOS
Field of view	84°
Nominal focal length	8.8 mm
Single image dimension (px)	5472 × 3684 × 3 (20Mb)
Number of images	423
Ortomosaic image size (px)	12779 × 17518 × 4 (426Mb)
Ground sampling distance (cm/px)	1.82
Mean flight height (m)	30

difficult task because variations during the image acquisition phase can result in multiple errors in the orthorectification process. Therefore, it is necessary to perform a post-processing quality evaluation and corrections to ensure that the reconstructed image presents an adequate representation of the real environment. For a better environment reconstruction, it is recommended to cover an area beyond the physical boundary of the interest area in question, making the borders and center areas to have higher image overlap, resulting in a better reconstruction per flight.

Most of our work was performed using the software Pynovisão (dos Santos Ferreira et al., 2017), using the link http://git.inovisao.ucdb.br/inovisao/pynovisao, it is possible to access the documentation.

We used the Pix4d (Pix4D, 2019) software for the orthomosaic creation and correction. The orthomosaic was generated using 423 images, considering longitudinal and lateral overlaps of 90% and 90%, respectively. The orthomosaic in Fig. 2(c) has a GeoTIFF format image with (11283 × 8671 × 3) pixels, that is a field representation of 3.2611 ha of the study site, each pixel has 3.33 cm^2. For optimal image processing a good geolocalization is necessary, it serves as a guide to the mapping software in the creation of the digital scenario. With this objective a total of six ground control points and sixteen checkpoints were distributed in the area. The Root Mean Square Error (RMSE) in the checkpoints were 7.1 cm in X, 4.6 cm in Y, and 9.5 cm in Z coordinates. The image mean projection error for the checkpoints were 1.38 pixels.

2.3. Manual object notation

We used Labelme software (https://github.com/wkentaro/labelme) to manually annotate trees in images, generating an archive that can be described as a ground truth of our desired area or a "mask". Fig. 3 (c) shows the generated mask.

2.4. SLIC superpixel

The machine does not see the images as we humans see, where we see colors, shapes, textures, and many other kinds of information. Instead, the machine "sees" significant conjunction of abstract binary data. To better differentiate and extract information from a given dataset, it is helpful for the machine that we somehow organize and make it simple. We used a superpixel algorithm for this task. The function of the superpixel is to group the pixels into perceptually meaningful atomic regions. There are many kinds of superpixels available to do that. Still, we choose the SLIC Superpixel, proposed by Achanta et al. (2012) since it can detect image boundaries, is simple to use, increase the speed, and improve the final quality of the result. The superpixel provides the machine with simplified information to do the classification tasks compared to the raw image. It organizes the data, separating (clustering) the data that is then interpreted as visual characteristics such as color, shape, and texture, reducing the computational complexity.

The feature extraction performed in this work takes into account 405 attributes: 36 for color, 214 for texture, and 155 for shape and gradient as described in Table 2. The attributes were based on previous approaches conducted by (Costa et al., 2019; dos Santos Ferreira et al., 2017) and consider now the improvement with the addition of the K-curvature extraction algorithm.

Superpixels-based methods aim to group pixels of a given image with similar or redundant characteristics such as color, texture, or shape into regions. The resulting dataset is less complex than the original pixel grid (S), making tasks such as object detection and object detection image segmentation more simple. There are many approaches to generate superpixels, varying in advantages and drawbacks, but the approach selected for our task was the SLIC method proposed by Achanta et al. (2012).

SLIC was found to be applicable to UAV orthoimages and feasible to accurately delineate object outlines taking into account the high-resolution provided by the UAV orthoimages (Crommelinck et al., 2017).While using SLIC, by default, the only parameter of the algorithm that we need to adjust is K, that is the desired number of approximately equally-sized superpixels, but we can also control the compactness (m). The greater the value of m, the more spatial proximity is emphasized and the more compact the cluster. This value can be in the range [1 to 20], and the default value for m in this paper is 10. The approximated size of a superpixel represents its complexity; the complexity O of the SLIC superpixel is dictated by the number of pixels in the image (N), resulting in the complexity O(N). For an image with N pixels, the approximate size of each superpixel is, therefore, N/K pixels. For roughly equally sized superpixels, there would be a superpixel center at every grid interval $S = \sqrt{N/K}$, that is the maximum spatial distance expected within a given cluster.

Following along in our research, we adopted three different base K numbers (K): 2000, 3000, and 4000. Although the K number generated for the tree canopies class was smaller than for the background class, since the area in the image representing the background was bigger, because of this disparity, we applied an under-sampling technique to balance the dataset in respect to the K distribution. Under-sampling is a technique used in data science that randomly reduces the K number of samples from the largest data set until it matches the K number of the smallest data set samples. The K number for each class before and after the K number under-sampling is presented in Table 3. The dataset used in this study can be accessed by the link: https://github.com/diegoandresantana/tree-canopies-experiment/.

2.5. Machine learning and classification

For the classification of the images, we used one Deep learning method and a set of five traditional Shallow Learning methods, as presented in Table 4. The shallow learners performed a supervised type of classification that depend on handcrafted features and instructions to target the problem. A more detailed description of the adopted classifiers is available in https://www.cs.waikato.ac.nz/ml/weka/ and all the parameters of the algorithms are set to their respective library default values. The classifiers used the numerical data and parameters extracted in the superpixel phase of our work and perform the classification of the orthomosaic. In this case, the classifier's objective is to learn what should be considered a tree canopies and what should be considered background, making it a binary classification.

2.5.1. ResNet50

The ResNet is a classic deep neural network winner of the ImageNet Challenge in 2015, and it was proposed by (He et al., 2016). The basic building block of a residual network is the Rectified Linear Unit (ReLU) activation function. The ReLU is responsible for transforming the

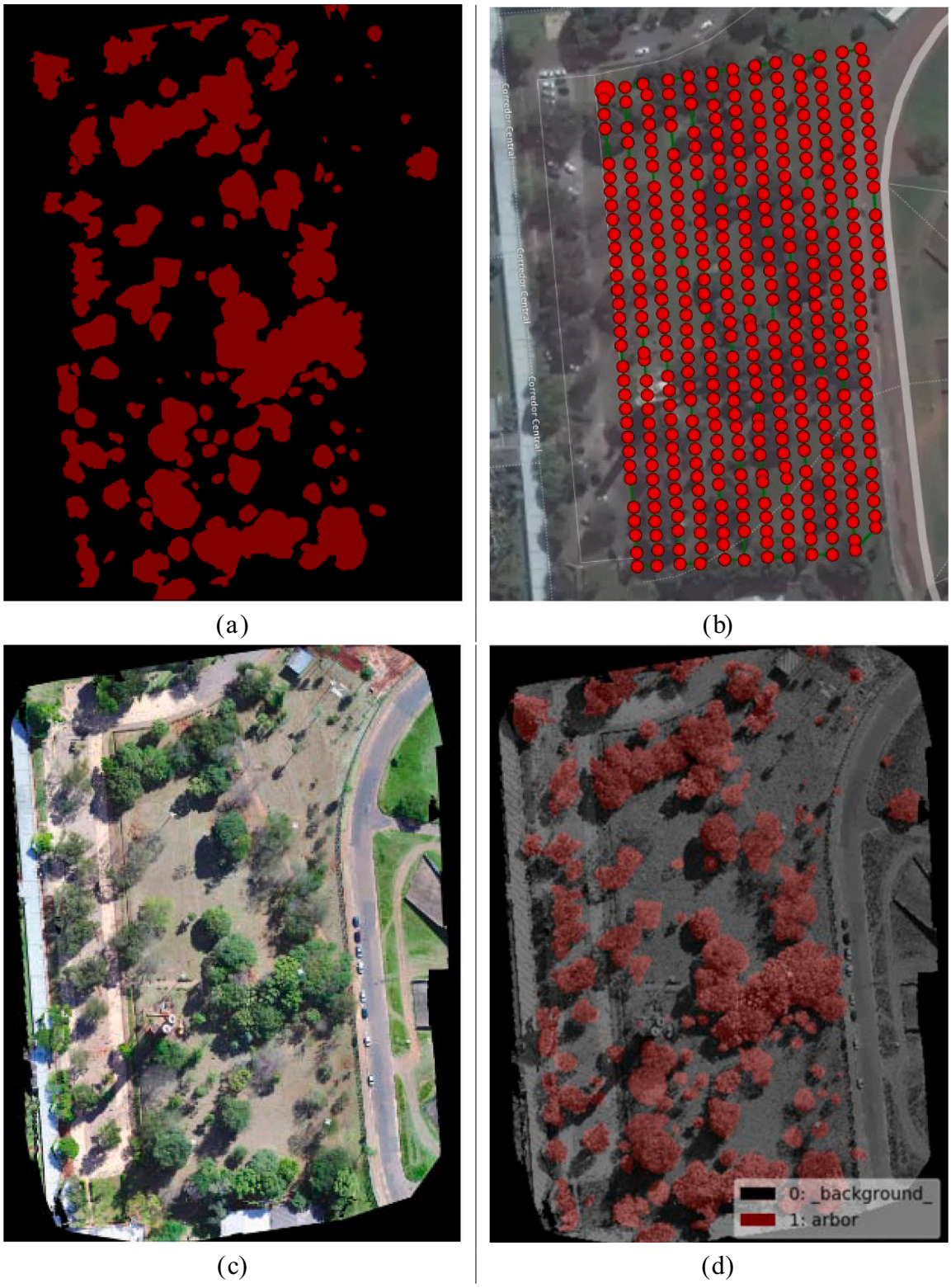

Fig. 3. (a) Mask generated with LabelMe Software using the orthomosaic. (b) Center location of each acquired image for the creation of the orthomosaic with lateral and longitudinal overlaps of 90% and 90%. (c) Orthomosaic of the study area. (d) Presentation of the overlap between the orthomosaic and the manual annotation.

summed weighted input from the node into the node's activation or output for that input in a neural network. It has become the default activation function for many neural networks because a model that uses it is often easier to train and may achieve better performance.

For our processing, stochastic gradient descent optimization was used, considering the learning rate of 0.01, a momentum of 0.9, and we also performed a variation in the batch size, adopting the configuration of 8, 16, and 32, respectively. It is important to mention that the variation in the batch number configures a new CNN. The number of epochs was 50, and all the weights were pre-trained, considering the Image-Net

Table 2
Description of the extraction methods and their respective quantity of extracted attributes.

Extractor	Method	Quantity	Reference
Color	RGB	12	Swain and Ballard (1991)
	HSV	12	
	CIELAB	12	
Texture	GLCM	36	Soh and Tsatsoulis (1999)
	GF	160	Feichtinger and Zimmermann (1998)
	LBP	18	Van Klaveren et al. (1999)
Shape and Gradient	Hu Moments	7	Hu (1962)
	Central moments	10	
	HOG	128	Triggs et al. (2000)
	K-curvature	10	Abu Bakar et al. (2015)

Table 3
Slic superpixel number corresponding to each Class for each total k number.

Slic superpixel number (k)	2000	3000	4000
Tree canopies	360	576	787
Background	1179	1837	2538
Tree canopies (balanced)	360	576	787
Background (balanced)	360	576	787

Table 4
Shallow learning and deep learning algorithms elected for the task.

Classifier	Reference
Random-Forest (RF)	Breiman (2001)
SVM	Platt (1999), Keerthi et al. (2001), Hastie and Tibshirani (1998)
J48	Salzberg (1994)
Naive Bayes (NB)	John and Langley (1995)
AdaBoostM1 (AM1)	Freund and Schapire (1996)
ResNet50	He et al. (2016)

weights (Deng et al., 2009). The output images consisted of resized superpixel segments of 256 × 256 × 3 pixels.

2.5.2. Performance metrics and validation

We calculated the confusion matrix for the evaluation of the classification, and from it, we derived the accuracy, precision, recall, and F1-score for the shallow and the Deep learning methods.

The accuracy is given by:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

where TP, TN, FP, FN stand for the number of true positives, true negatives, false positives and false negatives, respectively. In our analysis, positives and negatives refer to the pixels assigned by the underlying classifier to the Tree and Background classes, respectively. Such positives and negatives are true or false, depending on whether or not they agree with the ground truth.

The F1-score is given by:

$$F1 - \text{score} = 2 \times \frac{P \times R}{P + R} \quad (2)$$

where P and R stand for Precision and Recall, respectively, and are given by the ratios:

$$P = \frac{TP}{TP + FP} \quad (3)$$

$$R = \frac{TP}{TP + FN} \quad (4)$$

We also performed ten-fold cross-validation with the dataset using the accuracy metric as an evaluation. Cross-validation is a statistical method used to randomly split the dataset into training and test sets, ensuring that all folds have the same ratio of each superpixel class (Tree or background). The validation phase consists of picking one of the folders for the test. At the same time, the remaining folders were used to train the classifiers. This process is repeated ten times using each fold one time for the test set. As a result, the accuracy rate will be given by the average of the ten rounds of testing and validation. And finally, we processed an ANOVA test with the ten-fold cross-validation results for the accuracy metric.

The Union Intersect (IoU), also known as the Jaccard Index, is frequently used as a precision metric for semantic segmentation tasks (Wu et al., 2019; Berman et al., 2018). In the Reference and the Prediction mask, IoU is indicated by a ratio of the number of pixels in both masks to the total number of pixels, in the following equation we will present the formulation for the IoU:

$$\text{IoU} = \frac{|\text{Reference} \cap \text{Prediction}|}{|\text{Reference} \cup \text{Prediction}|} \quad (5)$$

3. Results and discussion

This section presents the results of the experimental evaluation of the selected semantic segmentation approaches. In Section 3.1 we present an evaluation in the terms of the metrics presented in Section 2.5.2, and we performed an analysis of the classifiers and the efficiency of each method. Then in Section 3.2 we performed a visual analysis of the results and presented the maps generated by the best classifiers.

3.1. Performance analysis

Tables 5 and 6 presented the results of the selected methods. The cross-validation results are highlighted in Table 7. Except for the NB, all the classifiers performed well in the task, achieving both accuracy and F-1 score above 85%.

Table 7 presents the average values for the accuracy achieved with the 10-fold, cross-validation with the adopted SLIC Superpixel configurations and methods. As we can see, the Superpixel and batch Numbers represent a decisive factor in the final result. For example, the best accuracy result for the 2000 K and 3000 K configuration are for the Random forest algorithm and for the 4000 K configuration the CNN with

Table 5
Metrics results for the shallow learning methods the best results are presented in bold.

Classifier	K-number	Accuracy	Precision	Recall	F1-score	Processing time seconds
RF	2000	0.9073	0.907	0.908	0.907	5.91
	3000	**0.9170**	**0.914**	**0.916**	**0.914**	10.71
	4000	0.9092	0.908	0.910	0.908	6.32
J48	2000	0.8668	0.863	0.864	0.863	3.75
	3000	0.8806	0.880	0.880	0.880	7.18
	4000	0.8633	0.870	0.872	0.871	3.87
NB	2000	0.7357	0.846	0.734	0.755	**0.72**
	3000	0.7372	0.847	0.737	0.756	1.14
	4000	0.7357	0.844	0.735	0.755	0.80
SVM	2000	0.9071	0.907	0.908	0.907	2.79
	3000	0.9131	0.909	0.911	0.909	7.33
	4000	0.9076	0.900	0.902	0.901	2.96
AM1	2000	0.8585	0.860	0.854	0.857	10.58
	3000	0.8631	0.867	0.860	0.862	16.87
	4000	0.8627	0.865	0.862	0.864	10.82

Table 6
Metrics results for the deep learning method with the variation in the batch size, the best results are presented in bold.

		ResNet50				
K number	batch size	accuracy	precision	recall	f1-score	Processing time h:m:s
2000	8	0.879	0.897	0.879	0.877	01:34:20
	16	0.856	0.879	0.856	0.851	01:30:00
	32	0.776	0.838	0.776	0.754	**01:27:00**
3000	8	0.885	0.898	0.884	0.883	02:22:30
	16	0.896	0.910	0.897	0.895	02:09:30
	32	0.907	0.917	0.907	0.906	02:06:40
4000	8	0.872	0.890	0.872	0.869	03:04:30
	16	**0.932**	**0.934**	**0.932**	**0.932**	02:51:40
	32	0.916	0.920	0.916	0.916	02:49:00

batch size 16, and this CNN result was also the best for all the methods analysed. We believe that this was because CNN models performs better the more data available; with more superpixels, the CNN had more data to learn, reaching better results, indicating also if that we presented the CNN with a higher K number it should perform better.

Fig. 4 presents a box plot of the performance obtained by each K number and each method separately. The range of variance obtained was smaller with the AdaBoost and Naive Bayes methods, but they performed worse than most of the classifiers used, so we omitted them to show more clearly the results of the other techniques, the Box-plot highlights that even when a CNN is selected, the processing configuration is critical. The classifier results were endorsed using the ANOVA analysis at a confidence level of 95% using the accuracy as a metric. The p-value was 2^{-16} meaning that we can reject the null hypothesis and evidencing a statistical difference in the performance of the methods tested. Latter in the test we will see that the while some networks performed well with the accuracy metric, they are a poorly indicator of the resulting maps, and a more suitable metric for analysing the maps is the IoU.

Although the big image size, low resolution, and complexity of the image impose a problem for most image processing algorithms, the deep neural networks still returned highly accurate results. The error can be explained by the qualitative phase of manual segmentation, which may also include the human error in manipulating the data. It is also important to mention that the boundary delineation is still a challenge for the machine learning field (Lecun et al., 2015; He et al., 2016; Goodfellow et al., 2016).

3.2. Visual analysis

Fig. 5 presents the SLIC superpixels processing output and what is considered as tree canopies (red) and background (green) using 2000, 3000, and 4000 superpixels configurations. We put the 3000 SLIC superpixel configuration in detaching because it resulted in a slightly better result for this dataset, as we will see in Section 3.2.1.

The dataset used in this work is characterized by various canopy distribution and many other features such as roads, cars, buildings illumination conditions, and so on, giving a lot of heterogeneous

Table 7
Results of the average of the 10-fold cross-validation performed using accuracy as a metric for all classifiers used. CNN represents the ResNet50, the B and the number following the CNN stands for batch size. RF represents the random forest, NB the Naive Bayes and AM1 the AdaBoostM1.

	Classifier average accuracy							
Super pixel number	J48	RF	SVM	NB	AM1	CNN_B8	CNN_B16	CNN_B32
2000	0.8668	**0.9073**	0.9071	0.7357	0.8585	0.8790	0.8560	0.7760
3000	0.8806	**0.9170**	0.9131	0.7372	0.8631	0.8850	0.8960	0.9070
4000	0.8633	0.9092	0.9076	0.7357	0.8627	0.8720	**0.9320**	0.9160

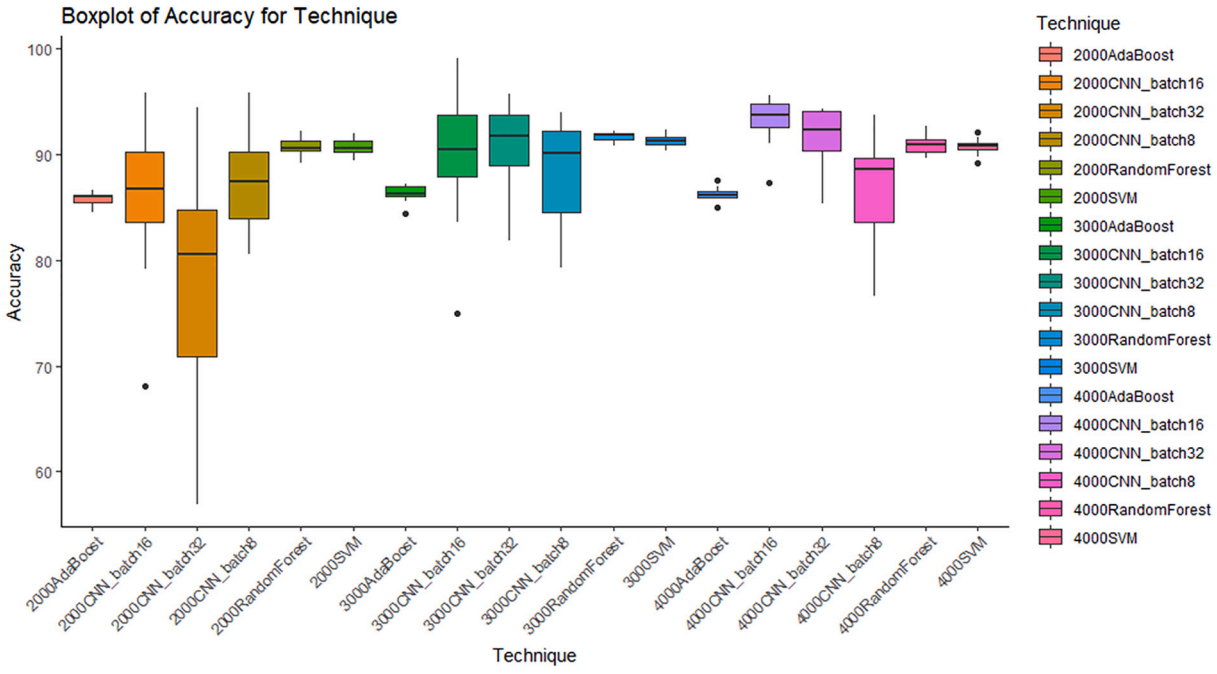

Fig. 4. Box plot comparing performance of classifiers for the cross validation accuracy metric. Because of the shortage of space, we take off the J48 and Naive Bayes results, because they performed poorly than the others.

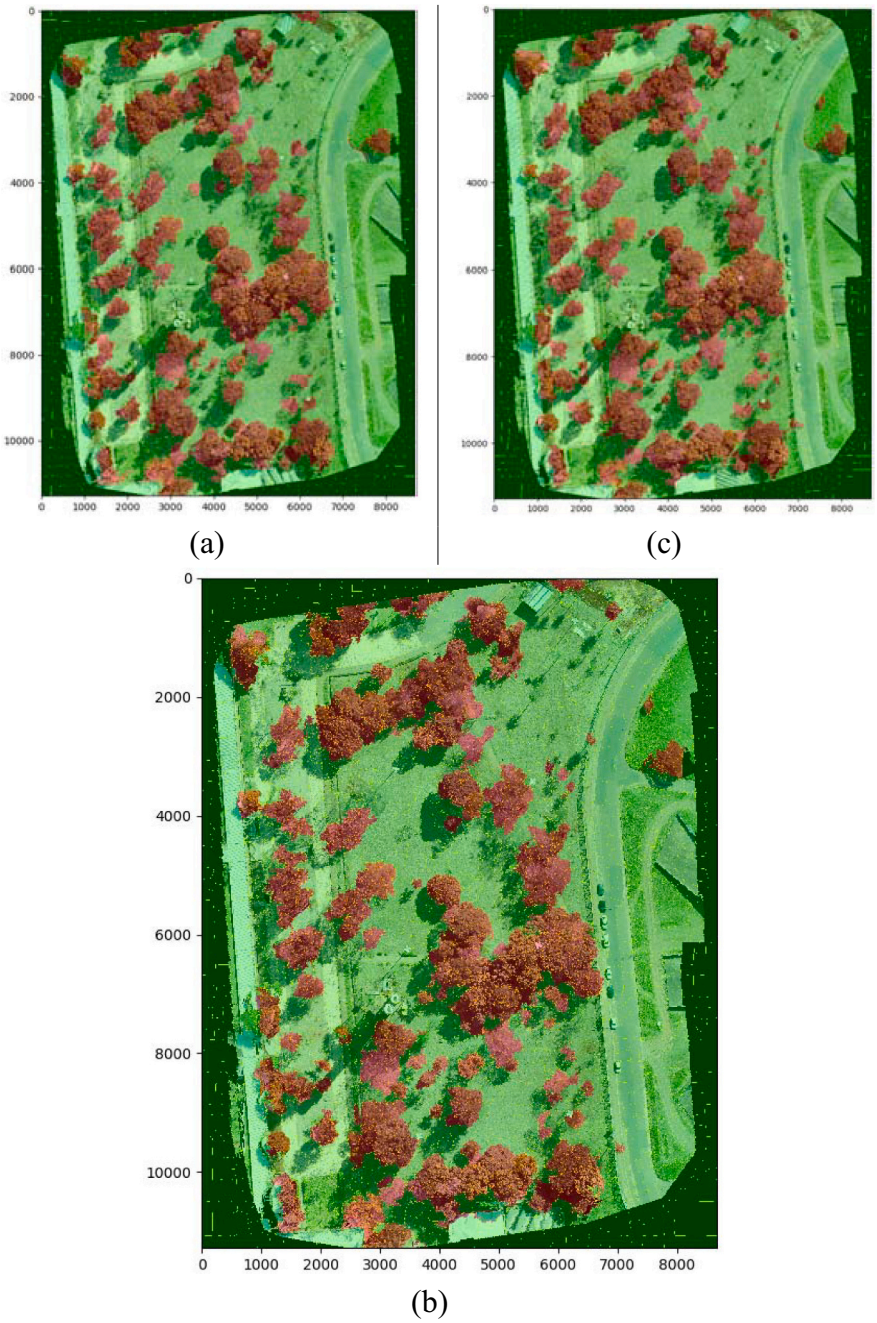

Fig. 5. Results of the SLIC superpixel segmentation divided by the K number: (a) 2000 K; (b) 3000 K; (c) 4000 K. The lateral and bottom numbers represents the size of the image in pixels, the red area represent the area where the superpixel classified the image as a tree canopy, and the green area represent the area where the superpixel classified the image as background.

information for the machine to learn, making it a complex segmentation task. Although differences in illumination geometry and the orientation of tree canopies should oppose a hindrance for the network to handle. In some regions, CNN classified as tree some parts outside the orthomosaic, making notable that the segmentation need to improve in the sense to avoid this. Some factors could explain this condition in the dataset: highly shadow-affected areas, a highly heterogeneous environment, the unbalanced dataset (originally much more samples for background than for the tree canopies), and, small dataset to learn, making it harder for the Superpixel and, therefore the CNN to learn what our object of interest is, remembering that our classifiers were all trained from scratch.

3.2.1. Resulting maps

In this section we present another phase of the analysis, that is creating maps with the models that we selected, the Maps are presented in Figs. 8–10, the performance visualization of results that we want to bring are presented in Fig. 2.5.2. In summary, the segmentation performance was satisfactory for the CNN's and bad for the shallow networks, despite the good accuracy results presented in Table 7. As we can see in Figures 8–10, the shallow networks returned maps that are far from reality, that cannot be used in real world applications.

For a better analysis of this phase we present these data in a quantitative manner in Tables 8 and 9. The area of the pixels as said before we had a total of 3.33 hectares of area, where 2.46 are for background and 0.79 are for tree canopies, our object of interest. We see that the best

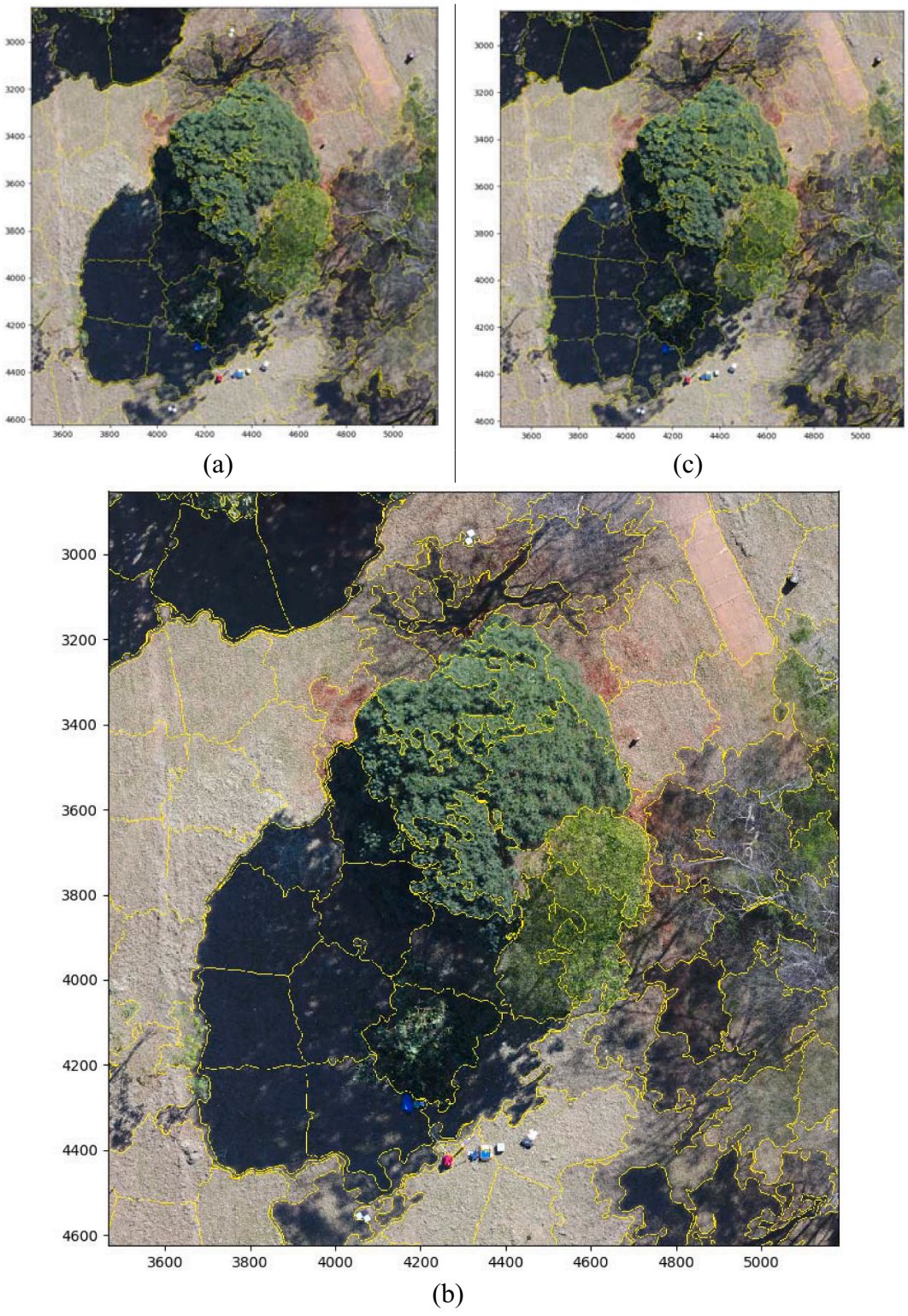

Fig. 6. A more detailed picture of the SLIC superpixel segmentation, presenting the segmentation, the yellow lines represent the border of each SLIC superpixel and each picture represent a different K number: (a) 2000 K; (b) 3000 K; (c) 4000 K.

Fig. 7. Guide to the visualization of the classification results. TN = True positives, FP = False positives, FN = False negatives and TP = True positives.

Result for the IoU metric where for classifier CNN K 3000 and B 16, followed by our best classifier in terms of accuracy CNN K 4000 B 32. But we see that neither of those classifiers are the closest in terms of difference in area estimated that was the classifier CNN K 3000 B 8, bringing only 1,01% of error when related to our interest object. This is a good result that can help a lot in real cases estimations for area, the big

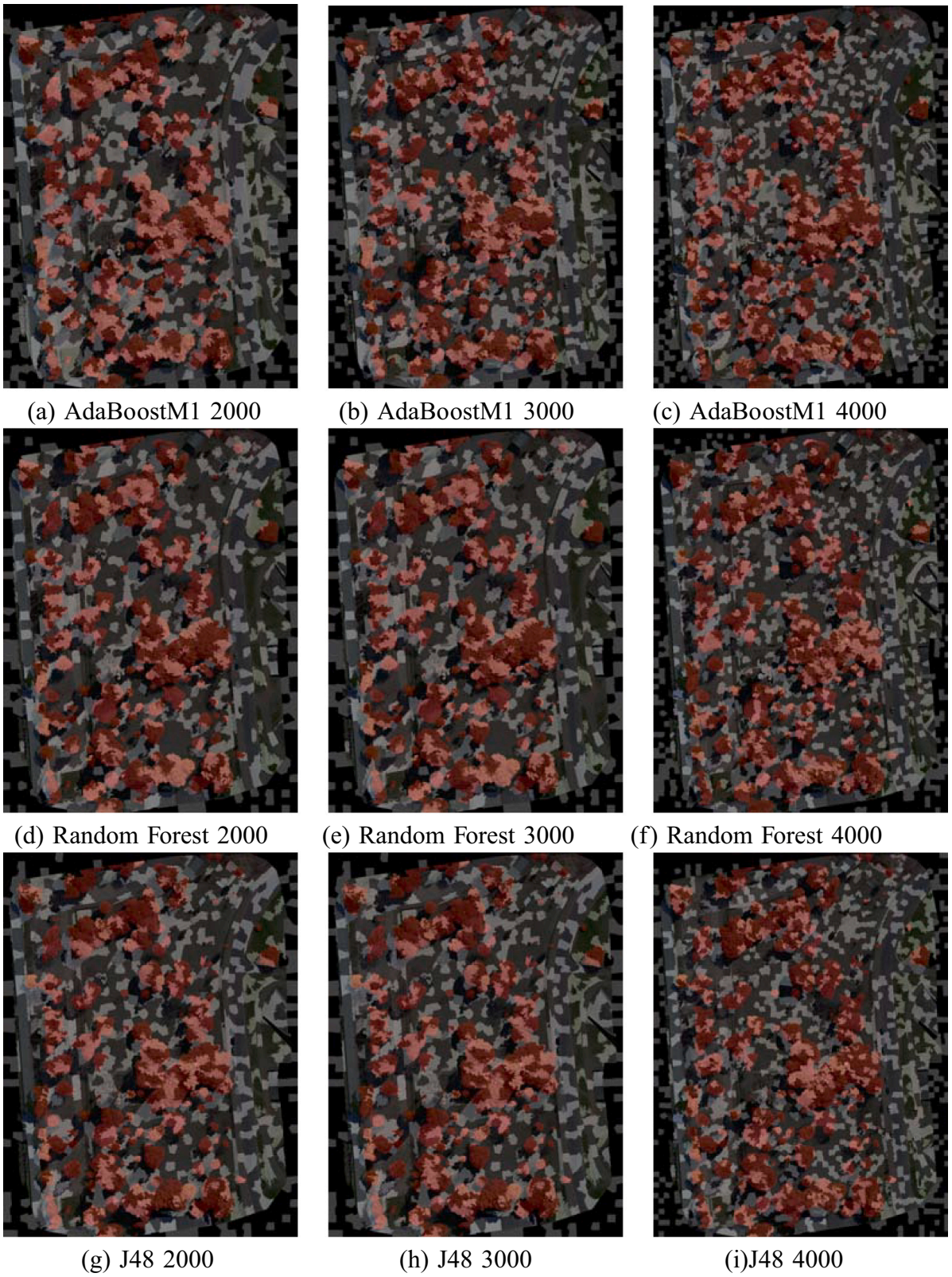

Fig. 8. Overlapped segmentation maps of the classifiers used in this research for AdaBoostM1, Random Forest and J48 with superpixels datasets of 2000, 3000, and 4000. (a) AdaBoostM1 2000. (b) AdaBoostM1 3000. (c) AdaBoostM1 4000. (d) Random Forest 2000. (e) Random Forest 3000. (f) Random Forest 4000. (g) J48 2000. (h) J48 3000. (i) J48 4000.

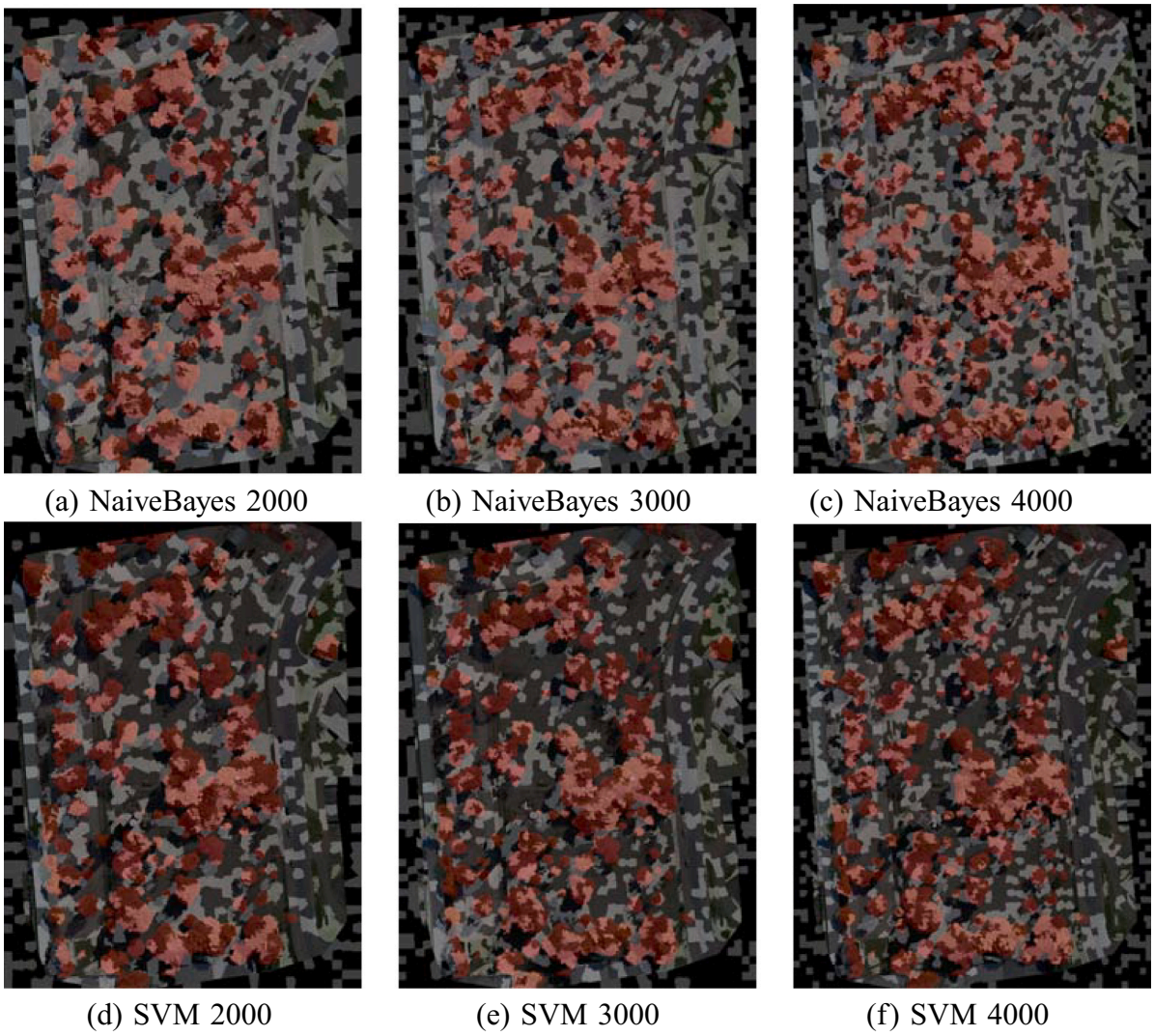

Fig. 9. Overlapped segmentation maps of the classifiers used in this research for Naive Bayes and SVM with superpixels datasets of 2000, 3000, and 4000. (a) NaiveBayes 2000. (b) NaiveBayes 3000. (c) NaiveBayes 4000. (d) SVM 2000. (e) SVM 3000. (f) SVM 4000.

difference for the error in area and the IoU, may be due to the characteristics of the superpixel, that is less smooth than an human made delineation.

The things to note here are that the best segmentation results achieved an accuracy higher than 93%, IoU of 0,700 and an error of estimation of three canopies area of 1%, suggesting that the proposed approach can target the proposed problems with precision.

4. Conclusion

This paper proposes a comparison of deep learning and shallow learning algorithms combined with the SLIC Superpixel method to recognize trees in urban environments using high-resolution images acquired with a consumer-grade UAV platform. The results presented support that the Deep learning methods presented in this research are capable of segmenting tree canopies inside complex urban environments. As presented, the optimum configuration of the processing, such as choosing the classifier and the number of parameters, is fundamental for the optimal final result. Our results show that SLIC used in conjunction with CNN's achieving high accuracy and IoU values and little error when estimating the area of canopies. We cannot say the same for SLIC and shallow learning techniques.

In terms of the quantitative results. We reach better accuracy results with the configuration with more superpixels, we estimate it to be because the CNN classifiers need more data to process and learn, making the dataset with more Superpixels a more suitable dataset for the CNN operate. Furthermore, considering the available publications, research studies that applied computer vision and machine learning techniques had better general results with CNN classifiers, in the detriment of the shallow methods, thus corroborating the results presented in this work, but, with the observance for the optimal parameters configuration. The big difference in results relay in the superior capacity of learning abstract data from the CNN's, the Urban environment is a complex dataset to be classified, with a lot of variation in colors, shapes and texture, that are emulated by the SLIC, thus creating a wide network, with 405 input layers in this case. The abstraction and estimation of weights in the Network need to have more layer or in another words, be more deep, to be capable of correctly back-propagate and update the weights of the network, thus reaching a better result in terms of IoU and area estimation, the data that interest us after all.

So for an estimation of the canopy area of trees that are present in an given area, even when we have 30% of mistake in the IoU we can have an very good estimation for area with just 1% in error when using a deep learning approach, suggesting it is capable of capturing this feature using only the SLIC data. Furthermore, the number of high-resolution and high-spectral images can be increased to detect more delicate

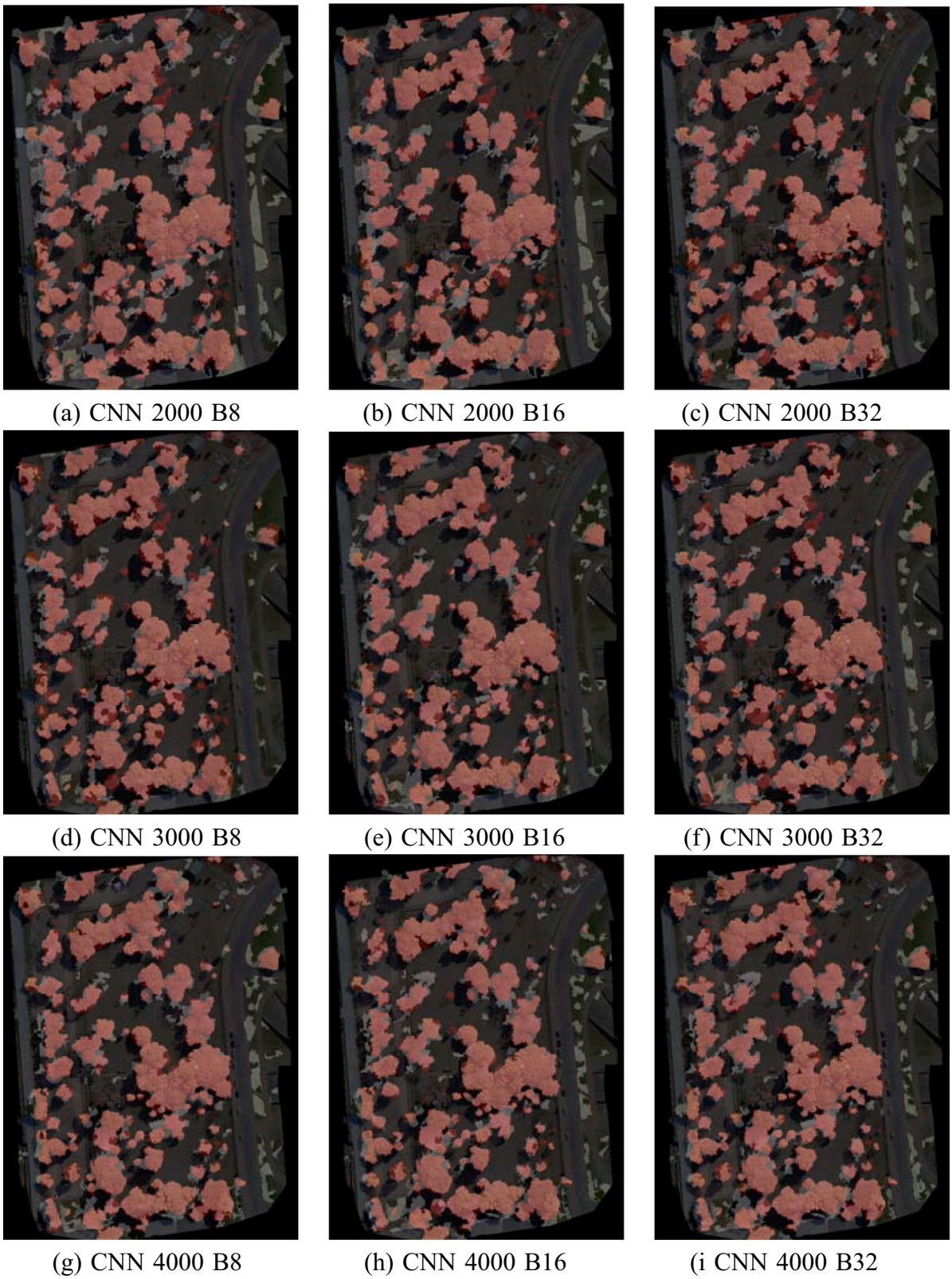

Fig. 10. Overlapped segmentation maps of the classifiers used in this research for CNN (ResNet50) an three batch-sizes (8, 16, and 32) with superpixels datasets of 2000, 3000, and 4000. (a) CNN 2000 B8. (b) CNN 2000 B16. (c) CNN 2000 B32. (d) CNN 3000 B8. (e) CNN 3000 B16. (f) CNN 3000 B32. (g) CNN 4000 B8. (h) CNN 4000 B16. (i) CNN 4000 B32.

Table 8
Performance results for the classifiers: Manual label (Control). CNN. AdaBoostM1 and J48. In the first row we have the classifier. In the second row we have the Labels Background or tree. In the third row we have the area of the label in hectares. In the Fourth row we have the difference for that classifier in relation to the manual label and in the fifth and last row we have the IoU metric.

Classifier	Label	ha	Difference %	IoU
Manual label	Background	2.462175	0.00%	1
	Tree	0.795726	0.00%	
CNN K 2000 B 8	Background	2.230537	4.94%	0.613
	Tree	1.027365	12.71%	
CNN K 2000 B 16	Background	2.318535	3.00%	0.644
	Tree	0.939367	8.28%	
CNN K 2000 B 32	Background	2.405856	1.16%	0.635
	Tree	0.852046	3.42%	
CNN K 3000 B 8	Background	**2.478018**	**0.32%**	0.680
	Tree	**0.779884**	**1.01%**	
CNN K 3000 B 16	Background	2.360387	2.11%	**0.700**
	Tree	0.897514	6.01%	
CNN K 3000 B 32	Background	2.395768	1.37%	0.657
	Tree	0.862134	4.01%	
CNN K 4000 B 8	Background	2.325495	2.85%	0.680
	Tree	0.932407	7.91%	
CNN K 4000 B 16	Background	2.321046	2.95%	0.686
	Tree	0.936855	8.15%	
CNN K 4000 B 32	Background	2.337523	2.60%	0.692
	Tree	0.920379	7.26%	
ADA K 2000	Background	1.980607	10.84%	0.200
	Tree	1.277295	23.23%	
ADA K 3000	Background	2.097711	7.99%	0.184
	Tree	1.16019	18.63%	
ADA K 4000	Background	2.133537	7.15%	0.168
	Tree	1.124365	17.12%	
J48 K 2000	Background	2.173015	6.24%	0.179
	Tree	1.084887	15.38%	
J48 K 3000	Background	2.206231	5.48%	0.164
	Tree	1.051671	13.85%	
J48 K 4000	Background	2.234437	4.85%	0.165
	Tree	1.023465	12.52%	

features better and improve the detection rate of any desired object, we also suggest a lower level of overlap, often while reconstructing the scene, a high number of data in the same place generate noisy mosaics, because trees are a dynamic object, that is, their leaves and branches are in constant movement due to the wind. The current methodology fails to identify single small trees and trees in very shadowed areas. The present approach does not compute the numbers of canopies inside a given area but the results allows the canopy area coverage calculation to be accurately estimated.

In future works we intend to use bigger and more detailed datasets to investigate the cerrado Biome. We can use these results to estimate biophysical parameters in the future. As future works, it is necessary to improve and refine the information extraction such as the number and diversity of species in a specific region and also sanity of determined region. For this, architectures and methods that enable the realization of instance segmentation, such as BlendMask, YOLACT, Mask RCNN, and Detectron2.

Funding

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001, FUNDECT, Foundation to Support Development of Education, Science and Technology of the State of Mato Grosso do Sul, Brazil and CNPQ (National Council for Scientific and Technological Development) through research grants codes: CAPES PRINT: 88881.311850/2018-01; 88887.580307/2020-00; 88887.635596/2021-00; CNPQ: 433783/2018-4, 314902/2018-0, 313887/2018-7, 436863/2018-9, 303559/2019-5, and 304173/2016-9.

Table 9
Performance results for the classifiers: Naive Bayes. Random Forest and SVM. In the first row we have the classifier. In the second row we have the Labels Background or tree. In the third row we have the area of the label in hectares. In the Fourth row we have the difference for that classifier in relation to the manual label and in the fifth and last row we have the IoU metric.

Classifier	Label	ha	Difference %	IoU
NB K 2000	Background	1.504077	24.16%	0.211
	Tree	1.753825	37.58%	
NB K 3000	Background	1.534198	23.22%	0.208
	Tree	1.723704	36.83%	
NB K 4000	Background	1.574593	21.99%	0.207
	Tree	1.683309	35.80%	
RF K 2000	Background	2.197762	5.67%	0.190
	Tree	1.06014	14.25%	
RF K 3000	Background	2.195431	5.73%	0.172
	Tree	1.062471	14.36%	
RF K 4000	Background	2.229294	4.96%	0.179
	Tree	1.028608	12.77%	
SVM K 2000	Background	2.182396	6.02%	0.177
	Tree	1.075506	14.95%	
SVM K 3000	Background	2.197824	5.67%	0.191
	Tree	1.060078	14.24%	
SVM K 4000	Background	2.2284	4.98%	0.184
	Tree	1.029502	12.81%	

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We would like to thank the Graduate Program of Environmental Technologies of the Federal University of Mato Grosso do Sul (UFMS) to support the doctoral dissertation of the first author. We would like to thank NVIDIA Corporation for the donation of the GPU used in this research. We also thank the editor and three reviewers for providing constructive concerns and suggestions. Such feedback helped us improve the quality of the manuscript.

References

Abu Bakar, M.Z., Samad, R., Pebrianti, D., Mustafa, M., Abdullah, N.R.H., 2015. Finger application using K-Curvature method and Kinect sensor in real-time. In: 2nd International Symposium on Technology Management and Emerging Technologies, ISTMET 2015 – Proceeding (January 2016), pp. 218–222. https://doi.org/10.1109/ISTMET.2015.7359032.

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2012. SLIC superpixels. IEEE Trans. Pattern Anal. Mach. Intell. https://doi.org/10.1109/TPAMI.2012.120.

Adhikari, A., Kumar, M., Agrawal, S., Raghavendra, S., 2021. An integrated object and machine learning approach for tree canopy extraction from UAV datasets. J. Indian Soc. Remote Sens. 49 (3), 471–478.

Baró, F., Chaparro, L., Gómez-Baggethun, E., Langemeyer, J., Nowak, D.J., Terradas, J., 2014. Contribution of ecosystem services to air quality and climate change mitigation policies: the case of urban forests in Barcelona, Spain. Ambio 43, 466–479. https://doi.org/10.1007/s13280-014-0507-x.

Berman, M., Triki, A.R., Blaschko, M.B., 2018. The lovász-softmax loss: A Tractable Surrogate for The Optimization of The Intersection-Over-Union Measure in Neural Networks. Computer Vision Foundation, United States. arXiv:1705.08790.

Breiman, L., 2001. Random forests. Machine Learning 45 (1), 5–32. https://doi.org/10.1017/CBO9781107415324.004.
Costa, C.S., Tetila, E.C., Astolfi, G., Sant'Ana, D.A., Brito Pache, M.C., Gonçalves, A.B., Garcia Zanoni, V.A., Picoli Nucci, H.H., Diemer, O., Pistori, H., 2019. A computer vision system for OOCYTE counting using images captured by smartphone. Agric. Eng. 87, 102017 https://doi.org/10.1016/j.aquaeng.2019.102017.
Crommelinck, S., Bennett, R., Gerke, M., Koeva, M.N., Yang, M.Y., Vosselman, G., 2017. SLIC superpixels for object delineation from UAV data. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, IV-2/W3. ISPRS, Germany, pp. 4–7.
Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. Imagenet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, pp. 248–255.
dos Santos, A.A., Marcato Junior, J., Araújo, M.S., Di Martini, D.R., Tetila, E.C., Siqueira, H.L., Aoki, C., Eltner, A., Matsubara, E.T., Pistori, H., Feitosa, R.Q., Liesenberg, V., Gonçalves, W.N., 2019. Assessment of CNN-based methods for individual tree detection on images captured by RGB cameras attached to UAVS. Sensors. https://doi.org/10.3390/s19163595.
dos Santos Ferreira, A., Matte Freitas, D., Gonçalves da Silva, G., Pistori, H., Theophilo Folhes, M., 2017. Weed detection in soybean crops using ConvNets. Comput. Electron. Agric. 134, 314–324. https://doi.org/10.1016/j.compag.2017.10.027.
Fassnacht, F.E., Latifi, H., Stereńczak, K., Modzelewska, A., Lefsky, M., Waser, L.T., Straub, C., Ghosh, A., 2016. Review of Studies on Tree Species Classification from Remotely Sensed Data, 186. Elsevier, Netherlands, pp. 64–87.
Feichtinger, H.G., Zimmermann, G., 1998. A banach space of test functions for Gabor analysis. Gabor Analysis and Algorithms. Birkh”auser Boston, Boston, MA, pp. 123–170. https://doi.org/10.1007/978-1-4612-2016-9_4.
Feng, Q., Liu, J., Gong, J., 2015. UAV Remote sensing for urban vegetation mapping using random forest and texture analysis. Remote Sens. 7 (1), 1074–1094. https://doi.org/10.3390/rs70101074.
Freund, Y., Schapire, R.E., 1996. Experiments with a new boosting algorithm. In: Proceedings of the 13th International Conference on Machine Learning doi: 10.1.1.133.1040.
Geron, A., Géron, A., 2017. Hands-On Machine Learning with Scikit-Learn & Tensorflow, 2. O'Reilly Media. arXiv:1412.3919.
Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep learning-whole book. Nature 1, 1–777. https://doi.org/10.1038/nmeth.3707.
Hastie, T., Tibshirani, R., 1998. Classification by pairwise coupling. Ann. Stat. 26, 451–471. https://doi.org/10.1214/aos/1028144844.
He, J.S. Kaiming, Zhang, X., Ren, S., 2013. Deep residual learning for image recognition. Front. Phys. 4 (April), 428–429. https://doi.org/10.3389/fpsyg.2013.00124.
He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. https://doi.org/10.1109/CVPR.2016.90.
Heinrich, G., 2016. Image Segmentation Using DIGITS 5.
Hu, M.K., 1962. Visual pattern recognition by moment invariants. IRE Trans. Inf. Theor. 8, 179–187. https://doi.org/10.1109/TIT.1962.1057692.
Jensen, R.R., Hardin, P.J., Bekker, M., Farnes, D.S., Lulla, V., Hardin, A., 2009. Modeling urban leaf area index with Aisa + hyperspectral data. Appl. Geo. 29 (3), 320–332.
John, G.H., Langley, P., 1995. Estimating continuous distributions in Bayesian classifiers George. In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. https://doi.org/10.1109/TGRS.2004.834800.
Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R.K., 2001. Improvements to Platt's SMO algorithm for SVM classifier design. Neural Comput. 13, 637–649. https://doi.org/10.1162/089976601300014493.
Kwong, I.H., Fung, T., 2020. Tree height mapping and crown delineation using LiDAR, large format aerial photographs, and unmanned aerial vehicle photogrammetry in subtropical urban forest. Int. J. Remote Sens. 41, 5228–5256. https://doi.org/10.1080/01431161.2020.1731002.
Lecun, Y., Bengio, Y., Hinton, G., 2015. Deep Learning, 521. Nature, London, England, pp. 436–444.
Ma, L., Li, Y., Li, J., Tan, W., Yu, Y., Chapman, M.A., 2019. Multi-Scale Point-Wise convolutional neural networks for 3d object segmentation from LiDAR point clouds in large-scale environments. IEEE Trans. Intell. Transpt. Sys. 22, 821–826. https://doi.org/10.1109/TITS.2019.2961060.
Noh, H., Hong, S., Han, B., 2015. Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE International Conference on Computer Vision. https://doi.org/10.1109/ICCV.2015.178 arXiv:1505.04366.
Onishi, M., Ise, T., 2018. Automatic Classification of Trees using A Uav Onboard Camera and Deep Learning. arXiv. http://arxiv.org/abs/1804.10390.
Osco, L.P., Ramos, A.P.M., Pereira, D.R., Moriya, é.A.S., Imai, N.N., Matsubara, E.T., Estrabis, N., de Souza, M., Junior, J.M., Gonçalves, W.N., Li, J., Liesenberg, V., Creste, J.E., 2019. Predicting canopy nitrogen content in citrus-trees using random forest algorithm associated to spectral vegetation indices from UAV-imagery. Remote Sens. 11, 2925. https://doi.org/10.3390/rs11242925.
Piazza, G.A., Vibrans, A.C., Liesenberg, V., Refosco, J.C., 2016. Object-oriented and pixel-based classification approaches to classify tropical successional stages using airborne high-spatial resolution images. GISci. Remote Sens. 53, 206–226. https://doi.org/10.1080/15481603.2015.1130589.
Pix4D, S.A.A., 2019. Pix4D. https://www.pix4d.com/.
Platt, J., 1999. Fast training of support vector machines using sequential minimal optimization. Advances in Kernel Methods – Support Vector Learning, 1. IEEE, Xiamen, China, pp. 997–1001.
Puliti, S., Ørka, H.O., Gobakken, T., Næsset, E., 2015. Inventory of small forest areas using an unmanned aerial system. Remote Sens. 7, 9632–9654. https://doi.org/10.3390/rs70809632.
Salzberg, S.L., 1994. C4.5: programs for machine learning by J. Ross Quinlan. Machine Learning. Morgan Kaufmann Publishers, Inc., 1993. https://doi.org/10.1007/bf00993309.
Shafri, H.Z., Hamdan, N., Saripan, M.I., 2011. Semi-automatic detection and counting of oil palm trees from high spatial resolution airborne imagery. Int. J. Remote Sens. 32 (8), 2095–2115.
Shelhamer, E., Long, J., Darrell, T., 2017. Fully convolutional networks for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39 (4), 640–651. https://doi.org/10.1109/TPAMI.2016.2572683 arXiv:1605.06211.
Shojanoori, R., Shafri, H.Z., 2016. Review on the use of Remote Sensing for Urban Forest Monitoring, 42. Arboriculture & Urban Forestry, Atlanta, United States, pp. 400–417.
Soh, L.-K., Tsatsoulis, C., 1999. Texture analysis of SAR sea ice imagery using gray level co-occurrence matrices. IEEE Trans. Geosci. Remote Sens. 37 (2), 780–795. https://doi.org/10.1109/36.752194.
Sothe, C., De Almeida, C.M., Schimalski, M.B., La Rosa, L.E., Castro, J.D., Feitosa, R.Q., Dalponte, M., Lima, C.L., Liesenberg, V., Miyoshi, G.T., Tommaselli, A.M., 2020. Comparative performance of convolutional neural network, weighted and conventional support vector machine and random forest for classifying tree species using hyperspectral and photogrammetric data. GISci. Remote Sens. 57, 369–394. https://doi.org/10.1080/15481603.2020.1712102.
Swain, M.J., Ballard, D.H., 1991. Color indexing. Intl. J. Comput. Vision 7, 11–32. https://doi.org/10.1007/BF00130487.
Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. https://doi.org/10.1109/CVPR.2015.7298594.
Tang, L., Shao, G., 2015. Drone remote sensing for forestry research and practices. J. For. Res. 26, 791–797.
Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W., 2000. Bundle adjustment – a modern synthesis vision algorithms: theory and practice. Vis. Algor.: Theor. Practice 1883. https://doi.org/10.1007/3-540-44480-7_21.
Van Klaveren, E.P., Michels, J.P.J., Schouten, J.A., 1999. The Orientational and structural properties of N_2 and N_2-ar solids at High pressure. Intl. J. Modern Phys. C 10 (02n03), 445–453. https://doi.org/10.1142/S0129183199000334.
Wan, L., Li, Y., Cen, H., Zhu, J., Yin, W., Wu, W., Zhu, H., Sun, D., Zhou, W., He, Y., 2018. Combining UAV-based vegetation indices and image classification to estimate flower number in oilseed rape. Remote Sens. 10, 1484–1502. https://doi.org/10.3390/rs10091484.
Wu, J., Yang, G., Yang, X., Xu, B., Han, L., Zhu, Y., 2019. Automatic counting of in situ rice seedlings from UAV images based on a deep fully convolutional neural network. Remote Sens. 11, 691. https://doi.org/10.3390/rs11060691.
Zhao, W., Jiao, L., Ma, W., Zhao, J., Zhao, J., Liu, H., Cao, X., Yang, S., 2017. Superpixel-based multiple local CNN for panchromatic and multispectral image classification. IEEE Trans. Geosci. Remote Sens. 55, 4141–4156. https://doi.org/10.1109/TGRS.2017.2689018.