Smart Agricultural Technology 3 (2023) 100092

Contents lists available at ScienceDirect

Smart Agricultural Technology

journal homepage: www.journals.elsevier.com/smart-agricultural-technology

A deterministic technique for identifying dicotyledons in images

Josué Leal Moura Dantas [*,a], André Riyuiti Hirakawa [b], Bruno Albertini [b]

[a] Federal Rural University of Amazonia, PA 275, km 13, Parauapebas, 68515-000, Pará, Brazil
[b] University of São Paulo, Av. Prof. Luciano Gualberto, tv 3, 158, São Paulo, 05508-010, São Paulo, Brazil

ARTICLE INFO

Keywords:
Feature extraction
Image classification
Image processing
Object detection
Pattern recognition

ABSTRACT

The identification of plants is often based on leaf recognition. *Ipomoea* spp., a dicotyledon weed present in sugarcane plantations, has unique vesiculated venation patterns that can be used in the recognition process. The uncontrolled plantation environment imposes challenges to leaf-based plant identification, such as overlap, light intensity, and occlusion. This work proposes a method for accurate and fast identification of leaves using Haar-like features, Fuzzy Logic, and Connected Components to differentiate monocotyledons and dicotyledons. Fuzzy Logic is used to define the template size for Haar-like features, combined with Integral Image concept to reduce processing time by lowering the arithmetic operation count. Our proposal was able to differentiate the target dicotyledonous leaves in an uncontrolled field image with more than one dicotyledon leaf. The obtained accuracy is acceptable regarding the current literature and the processing time was reduced.

1. Introduction

Image processing has been widely used in agriculture for automated tasks, reducing costs, and improving the quality of processes. Robotics uses image processing extensively: counting of fruits per tree, as done by the robot Shrimp [1]; weed detection in a cotton plantation using images from a hyperspectral camera equipped on Ladybird [2]; and AgBotII using images to identify weeds and remove them [3].

Brazil is one of the largest sugarcane producers in the world, being responsible for 25% of global sugarcane production. As in other crops, weeds cause losses in sugarcane production. Among the sugarcane weeds, *Ipomoea* spp., which is a climbing weed that grows up over sugarcane crops, is one of the most difficult to remove, causing losses of 46% at highly infested areas [4]. Nowadays, sugarcane weeds are eliminated using pesticides, machines, or hand-weeding. Organic crops can only use light machinery or hand-weeding, mostly because the remaining options could damage the soil biota or are prohibited. Image processing supports the process of removing these weeds by precisely locating them, e.g. by aerial imagery, enabling the removal by light machinery (e.g. robots), or reducing workers' exposure to the harsh environment when using hand-weeding.

However, image processing in real crop environments presents a series of challenges. Brightness, occlusion, and overlap are some of the challenges found in these environments [5]. The objective of this research is to propose a method that can be applied in those environments, and still sustain a low response time and high accuracy when compared with similar techniques.

Weed recognition by imagery can be reduced to a classification problem for our target species, because *Ipomoea* spp. is a dicotyledon and sugarcane is monocotyledonous. This difference between the species of plants is noted, among other characteristics, in the aspect of the leaves. Sugarcane has a leaf with parallel veins, whereas the *Ipomoea* spp. has a reticulated vein. The research focused on finding a method to distinguish these two leaf patterns.

The venation of the leaf has been one of the main characteristics used to identify plants, as described by Codizar and Solano [6]. However, the contrast between the leaf and its venation is not easily identifiable, thus it is common to preprocess the image to extract this feature. Artificial Neural Networks [7] and Support Vector Machine (SVM) [5] are among the most successful techniques for this task, but require robust hardware for fast image processing in real-time.

Hough Transform is also used to identify the plant venation [8]. Hough Transform uses sinusoids in numerous directions to detect the lines at a specific point. However, this technique requires a previous conversion of the input image to grayscale, followed by an edge detection (often using Canny method) to finally apply Hough Transform. This strategy requires considerable processing time.

Deep Learning is used by Tan et al. [9] to classify the plants' leaves. Their proposal makes use of four main steps, which are: sampling, image preprocessing, feature extraction, and classification. In the first step, leaf

[*] Corresponding author.
E-mail addresses: josueleal@usp.br (J.L. Moura Dantas), balbertini@usp.br (B. Albertini).

https://doi.org/10.1016/j.atech.2022.100092
Received 3 March 2022; Received in revised form 29 June 2022; Accepted 30 June 2022
Available online 1 July 2022

samples were collected, and images were acquired. The leaves' images were then preprocessed and submitted to the feature extraction step to select and retrieve relevant information from the leaves using a Convolutional Neural Network (CNN) and Sobel edge detection approach. Lastly, the extracted features were trained and classified by using several machine learning methods. Using neural networks and Flavia dataset, it achieved a rate of 94.63%.

Zhang [10] propose a methodology they called SKEDET to obtain the leaf skeleton or venation. The proposed algorithm takes each white pixel after the segmentation process and draws a line for each angle (from 0 to 360 degrees). Then it uses a threshold to select the lines that have more pixels. Finally, it uses the longest line found, such as the central vein, and analyzes those that are connected to perform the classification. Tests ran using their image dataset with 44 images of the swiss potato plant containing a total of 826 leaves and obtained 87.78% accuracy.

Arrasco et al. [11] also use the classification of plants using venation. The methodology applied in their work consists of using Otsu technique for segmentation followed by the Prewitt operator to identify the venation of the leaf. The method was evaluated using a dataset built by the researchers, which contains 200 images of 10 different species from the Peruvian Amazon. Two classifiers are used for testing, the Linear Discriminant Analysis (LDA) and the Multilayer Perceptron (MLP). The best result was obtained by the MLP, with 87.75% of accuracy.

Kolivand et al. [12] performed the classification of plants using the leaf vein pattern. For this, they use the Canny technique for vein segmentation and extraction, followed by a rearrangement in a scatter point curve. Then HUE normalization is applied to increase the contrast between leaf and leaf venation. Finally, the image is analyzed for the identification of which curves are leaf veins and which are misdetection. Using the Flavia Dataset, the proposal obtained an accuracy of 98.60%.

The leaf textures are used by a robot to detect weeds in a sugarcane plantation in Sujaritha et al. [13]. A real-time fuzzy classifier based on leaf texture features is used on the robot. The proposed system detects weeds with 92.9% of accuracy while keeping the processing time bellow 20ms. However, the technique proposed by these authors assumes that there is only one plant species in the image.

Most of the works described above showed techniques that make use of a big amount of data to train the algorithm and the processing time is not considered as an issue. In most of cited articles, such as in Sujaritha et al. [13], many leaf characteristics were considered to recognize the plant, which implied a high computational cost. Others work utilized techniques such as the Hough Transform and SKEDET that also results in a considerable time to process the image.

Taking this into account, the method proposed in this paper is to analyze vein patterns using only the venation feature, extracted by Haar-like features. The technique allows the reduction of the parameters of an image by matching simple templates. The Integral Image concept also is used in this work to reduce the processing time, since it stores an intermediate image representation using summed areas. The Fuzzy Logic was used to define what would be a rectangle with a small, medium, or a large number of green pixels to define the most appropriate template size and classify the area as belonging to a dicotyledon or monocotyledon leaf, given the number of template occurrences in a region. The Connected Components technique was used to find the biggest exposed leaf on the image, increasing the correct recognition rate. Our major contribution is the use of Haar-like features, Fuzzy Logic, and Connected components for identifying plants using a venation pattern with low processing time and high accuracy in an uncontrolled environment.

2. Recognizing the leaf pattern

Our proposal to recognize the pattern embedded into dicotyledon leaves is divided into 5 steps, arranged on a pipeline, as depicted in Fig. 1. The pipeline arrangement can benefit from multitasking, which improves response time.

The **image** capture step is out of scope and could be any modern RGB camera output. In the next step, the image is analyzed considering its color space, size, and the possibility of size reduction, what is called **preprocessing** process, whose purpose is to verge the venation visibility on a binary image. The third step is **segmentation**, where the Canny technique is applied to separate the venation from the other parts of the leaf. At the **processing** phase, our predefined templates are used according to the specified **Haar-Like** features over the Integral Image. Following, the **Fuzzy Logic** and the concept of Connected Components (**CC**) are applied, producing the **output** image with annotated dicotyledonous leaves as red rectangles.

The color space is transformed in the preprocessing phase to improve the differentiation between the venation and other leaf areas, a frequent task in image recognition, such as leaf recognition [14]. Another usual preprocessing performed is the reduction of the image size to reduce the processing time. The next step to be taken is segmentation. Our preprocessing phase transforms the color space and normalize the image size, often reducing it. Those tasks reduce the processing time and are well known in image processing [14]. Our segmentation uses Canny [15] method to edge detection, a technique that uses a gradient descent that identifies sudden changes in pixel intensity values. Analyzing the output image of the segmentation phase, we decided to use 30.0% of the original image, because that was the lowest value found experimentally in which the venation patterns are visible enough without jeopardizing the detection accuracy.

The segmented image is scrolled by the templates defined according to the specific Haar-Like features, defined considering the relation of the intersections between the venation, which is the key difference between monocotyledonous and dicotyledonous: the parallel venation without intersections in the structure of the vein, present only in monocotyledonous. The initial set of templates were defined as shown in Fig. 2, using a naive approach.

The naive set of templates used presented two disadvantages. The

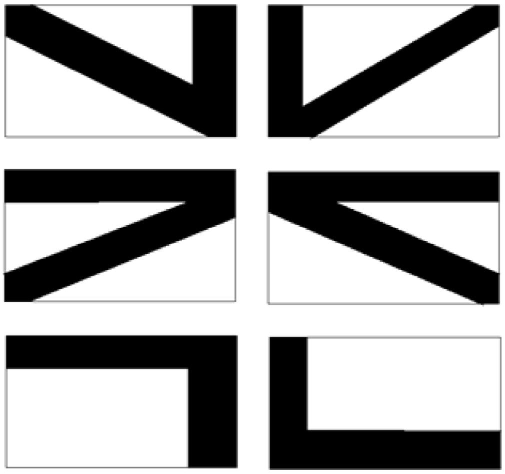


Fig. 2. Naive set of templates.

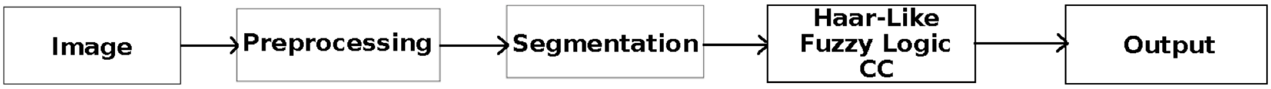


Fig. 1. Plant recognition pipeline.

first is that the intersections between the leaf vein lines are not regular as shown in the template image. The second is the difficulty of applying the concept of Integral Image when using those templates, because generally the concept is more easily applied in areas with two well-defined and parallel regions with gradient difference between them. The naive template set requires an amount of arithmetic operations that would cancel the speedup provided by this technique. As a result, and to solve the problems mentioned, the set of templates shown in Fig. 3 was defined.

The processing steps described above were implemented using the OpenCV 4.5.1 library (Open Source Computer Vision Library) with C++. The OpenCV is a widely used library for image processing. It is available as BSD license and is free for academic and commercial use. In addition, it is compatible with several Operating Systems, such as Windows, Linux, Mac OS, iOS, and Android [16].

In the following sections, the concepts of Haar-like features, Integral Image, Fuzzy Logic, and Connected Components are described in the context used in this work.

3. Materials and methods

3.1. Haar-like features

The Haar-like features technique consists of scrolling templates through an image, usually rectangles that have two or more black and white regions. Haar-like has been used in several applications, one of which is face recognition. The method was first described by Viola and Jones [17] for face recognition along with Adaboost, a machine learning technique. Their proposal aims to reduce the processing time of the images by reducing the space of features to analyze. Li et al. [18] used a similar approach to identify pedestrians, and Mohamed et al. [19] also used the Haar-Like features, as a descriptor, to classify electrocardiogram images.

At each scrolling interaction, the average of pixel values in the white region(s) are subtracted from the average of black pixels region(s), as shown in Eq. (1), where *wpv* is white pixels values, *nw* is the number of white pixels, *bpv* is black pixels values and *nb* is the number of black pixels. If the total of a template region is equal to 255, the template is found to be a complete correspondence for that location. If all pixels in a region are white the average is 255, and if all pixels are black, the average is 0, so the difference between the two regions is 255.

$$total = \frac{\sum wpv}{nw} - \frac{\sum bpv}{nb} \tag{1}$$

The difference between the regions can be used to find edges, lines, and other simple structures in the image [17]. It is possible to reduce the amount of information required to carry out the classification of an image by applying this method, which directly reduces the processing time and thus the response time.

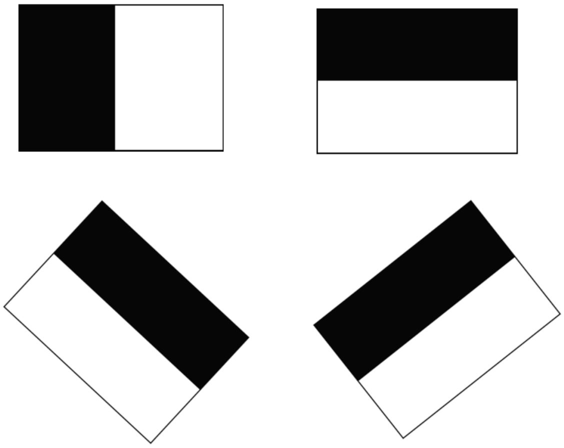


Fig. 3. Templates applied to find leaves.

3.2. Integral image

The Integral Image is an intermediate representation of the image used for faster processing. The pixels above and to the left side of a given position (x,y) are summed together with the value stored in the position by the following equation:

$$ii(x,y) = \sum_{x' \le x, y' \le y} i(x',y') \tag{2}$$

where *ii* is the integral image, (x,y) is the position on the integral image, *i* represents the original image, and (x',y') is the point position on the original image. Taking a specific point in an integral image, $ii(x,y)$ is the sum of each neighbor point (cross pattern), and the point value itself as in the original image.

In Eq. (3), $s(x,y)$ represents the cumulative row sum until the (x,y) point, $s(x,y-1)$ the cumulative row sum until the previous point in the same row, and $i(x',y')$ is value in the original image corresponding point. The result is an intermediary sum accumulated of the row until the specific point. Equation (4) uses the cumulative row sum point $(s(x,y))$ to calculate the integral image value using the same approach: $ii(x-1, y)$ is the accumulated integral image row point value until the previous row.

$$s(x,y) = s(x,y-1) + i(x',y') \tag{3}$$

$$ii(x,y) = ii(x-1,y) + s(x,y) \tag{4}$$

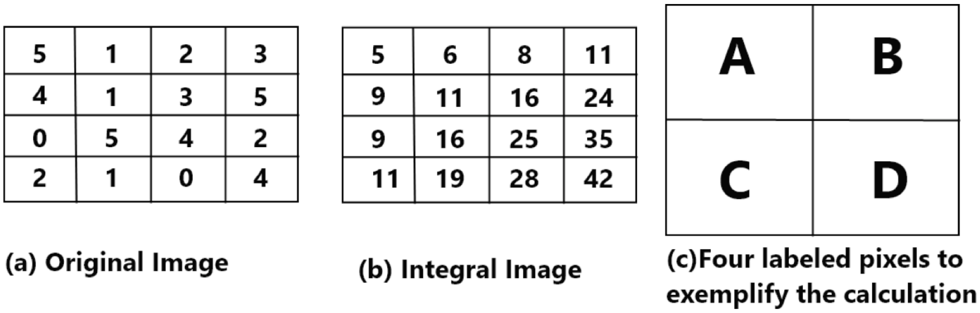
Figure 4 exemplifies how the Integral Image calculation is performed. Fig. 4(a) shown the original pixel values of the image, Fig. 4(b) the calculated value of the integral image, and Fig. 4(c) is an illustrative scheme of how the calculation is performed. In this example, to calculate the value of the *D* region, it is necessary to sum $C+B+D$ region values and decrease by the value of the *A* region, so the equation will become $C+\ B+\ D-\ A$. The calculation for (2,2) selects the values already calculated, as depicted at the right matrix: $(1,2)\ = 6$, $(2,1)\ = 9$, and $(1,1)\ = 5$, and the value of the point (2,2) itself in the original matrix, which is 1. The calculation is thus: $(2,2)\ = (2,1) + (1,2) + (2,2) - (1,1)$, so the result is 11, as can be seen in position (2,2).

The main advantage of using the Integral Image is the reduction of operations required by the calculation. For example, to calculate the cell value (4,4) of the Integral Image, four accesses to the matrix are used. If all cells were added together to find the value in the matrix, 16 accesses are required. An Integral Image is an intermediate image representation which main advantage is perceived when subsequent operations must perform a series of pixels-area addition, as in the case of Haar-Like features.

Suppose that a 16x4 template window is used to scroll through an image. Dividing the template in half, one for the black region and another for the white region, we would have two regions of size 16x2. This would require 32 accesses to the matrix to calculate the value in each region adding all pixels. The Integral Image concept can reduce this amount to 4 access, substantially improving the processing time.

3.3. Connected components

The labeling of Connected Components is widely used in the context of image processing and pattern recognition, as noted by He et al. [21] and Bataineh et al. [23]. This technique is used to identify the pixels belonging to a given object, using them to generate a binary image. Bataineh et al. [23] mentions the usage of this technique in several applications in the field of computer vision, involving the recognition of a target, recognition of license plates, analysis of medical images, recognition of faces and fingerprints, among others.



5	1	2	3
4	1	3	5
0	5	4	2
2	1	0	4

(a) Original Image

5	6	8	11
9	11	16	24
9	16	25	35
11	19	28	42

(b) Integral Image

A	B
C	D

(c)Four labeled pixels to exemplify the calculation

Fig. 4. Example of calculating the integral image [20].

He et al. [21] presents a definition, which he calls classic, of the process of labeling related components. Given a binary image of size NxN, the coordinates of a pixel (x,y) of the image is called $b(x,y)$, where $0 \leqslant x \leqslant N-1$ and $0 \leqslant y \leqslant N-1$. Foreground pixels are also called object pixels. While not stated otherwise, the values of object pixels and background pixels are assumed 1 and 0, respectively. Moreover, for convenience, all pixels in the border of an image are assumed background pixels.

For analyzing the connectivity of an object, 4 or 8 pixels neighboring of a central pixel are used. The 4 and 8 neighborhoods of a pixel are represented in Fig. 5, where the even P_n represents 4 neighbors positions and $P1$ to $P8$ are the 8 neighbor representation. In general, 8-connectivity is used for Connected Components (foreground pixels) and 4-connectivity for holes (background pixels) [21]. In summary, the technique's goal is to analyze the neighbors of a pixel looking for a certain value recursively to find related pixels and associate them to the same object. Any data structure can used to store the pixels that belong to the same object on the image.

3.4. Fuzzy logic

Fuzzy Logic, among other applications, can also be used in the image classification process. Li et al. [22] used decision trees along with the concept of Fuzzy Logic to perform automatic image annotation. Juneja and Rana [24], on the other hand, used the Gabor filter, Fuzzy Logic, and SVM for the classification of facial expressions.

Liu et al. [25] used Fuzzy Logic to enhance microscopic images of Chinese herbal medicines. In the methodology defined by the authors, a grayscale image is generated and each pixel is related to a gray level using a pertinence function, with a 50% threshold. Depending on the gray value, a correction in the image degree of gray is made.

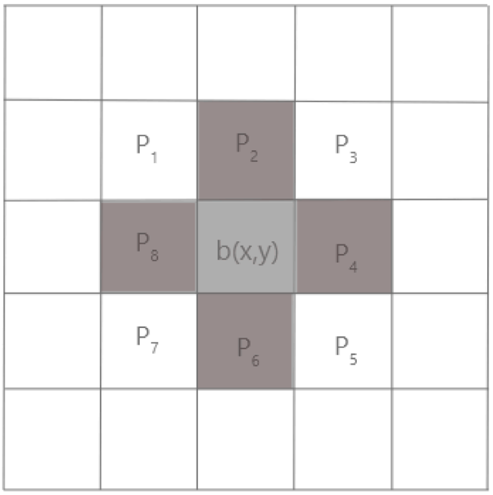


Fig. 5. 4 and 8 neighbors pixels representation.

We decided to use a fuzzy system to define what would be a square with small, medium, or large numbers of green pixels for the template size and classification of the area as belonging to a dicotyledon or monocotyledon leaf. We defined three variables to be used in the fuzzy inference system: the percentage of green pixels in the area, template size, and the total of template occurrences.

4. Results and discussions

Since our proposal targets uncontrolled environments, we captured images at a commercial farm. Their business is related to organic sugarcane plantations, so they often use hand-weeding.

We have defined an image scanning procedure, to define regions that should be analyzed due to the greater probability of containing a weed leaf. For each pixel, if the value of the green component is greater than the blue component and at least 90% of the value of the red component, the pixel is defined as belonging to the area of interest. These values were defined by analyzing the pixel values in the images, considering the soil and sky regions as exclusion candidates.

The square size was defined experimentally to 2.5% of the image size. We conducted an experiment using various square sizes, counting the number of squares with a green pixel majority, and squares with 2.5% of the image size produces the optimal relation between the the area selected for analysis and the area that can be ignored (on average, 46.0% of the image).

The division of the image area into squares allowed the application of templates of different sizes depending on the number of green pixels existing in that area. Bigger green areas imply bigger template size for Haar-like algorithm, decreasing the processing time due to the concept of Integral Image.

We used fuzzy triangular function due its low complexity and the binary scenario: venation is either a monocotyledon or not. We defined 3 linguistic variables, one for each classification problem: green level (f_G), template size (f_T), and monocotyledon binary classifier (f_C).

For green presence, our linguist values are low, medium and large, and we did not get any improvement adding more levels. We ran the classification over our dataset and grouped the squares to define f_G according to the green area percentage, extrapolating to the functions show in Fig. 6. For example, if an area has 10% green pixels, looking at the graph we notice that $f_G(10) = 1$. If the area has 20% green pixels, $f_G(20) = 0.1$ for medium, $f_G(20) = 0.9$ for low and $f_G(20) = 0$ for large.

The same approach was used to find the function or the linguistic variable representing the template size. We also did not get any improvement using more than the classical three possible values: small, medium, and large. For example, if a template size is 30, we have $f_T(30) = 1$ for large, and $f_T(30) = 0$ for medium and small.

An image of a monocotyledon leaf in a controlled environment would match just one template since the venation is parallel with no intersections, thus the variable values are low or high, which is show on Fig. 7. Regarding the membership function, if there is a correspondence of the same template 5 times in the same area of interest (the green areas), the membership function assumes the following values: $f_C(5) =$

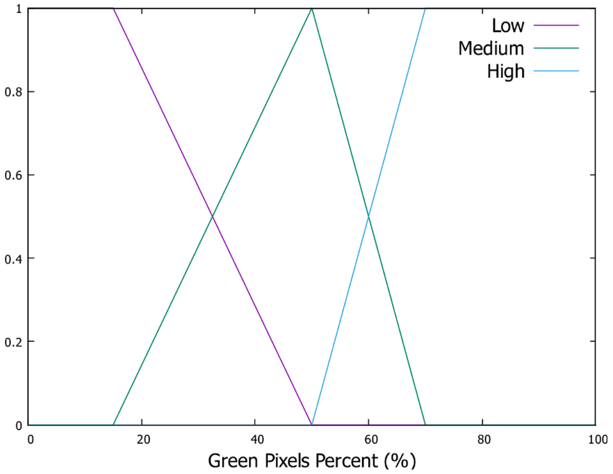


Fig. 6. Pertinence function referring to the percentage of green pixels existing in a given area of the image. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

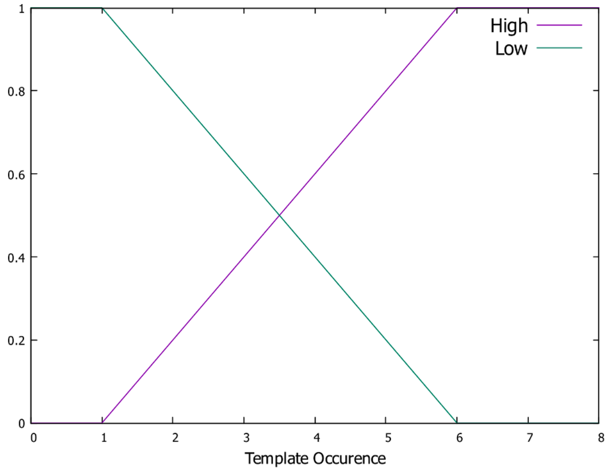


Fig. 7. Pertinence function that represents the number of occurrences of a template.

0.4 corresponding to the low occurrence and $f_C(5) = 0.6$ for high occurrences.

To define which template size to use, we entangled f_G and f_T. Using the green percentage we determine the f_G and does the reverse procedure on f_T to determine the most suitable template size for that region. This imply solving $f_G(i) = f_T(j)$ for a given i. For example, if the green pixel percentage is 60%, the membership function results $f_G(60) = 0.5$ for large, $f_G(60) = 0.5$ for medium and $f_G(60) = 0$ for small. Analyzing the graph in Fig. 6, the value 60 on x axis, green pixels percent, intersects the value on the small line on 0.0 on y axis, and 0.5 value on y axis for medium and large lines. The closest $f_T(j)$ is $j = 25$, $f_T(25) = 0.5$ for large, $f_T(25) = 0.5$ for medium, and $f_T(25) = 0$ for small, thus the template size used on a 60% green area is 25.The graph in Fig. 8 shows the value 0.5 on y axis intersects the medium and large line corresponds to the point 25 in the x, size of the template.

We selected 146 images (2448x3264 pixels, RGB colorspace, JPEG) to test our proposal. The processing time for each image was 0.146 s in a desktop computer (Intel Core i7-8565U CPU, 1.8GHz, 16GB) without any GPU acceleration. The accuracy was calculated based on the ratio of the weed leaves recognized and the total of leaves in the images. The time to processing an image is 146 ms.

Most of the previous works about image weed recognition used databases of images taken at a controlled or semi-controlled (e.g. aided by artificial light) environment. This often results in an artificial high accuracy, since their performance probably will not sustain when applied to real environments. Academic literature seems to consider any accuracy over 85% as acceptable for real environments, and any performance beyond that is considered a success. For our work, over controlled environment the accuracy was 95%, and for uncontrolled environment using random parameters (e.g. occlusion, different luminosity, etc.) we reached a general accuracy of 81.6%. Although we could find techniques reaching 95% [9], there is no data regarding real uncontrolled environments. During our search, the most similar work that we could find was Sujaritha et al. [13], which is very similar to the context for which we developed our proposal, reaching 92.9%, but only works if there is one species on the image, what we consider a hard restriction for real application.

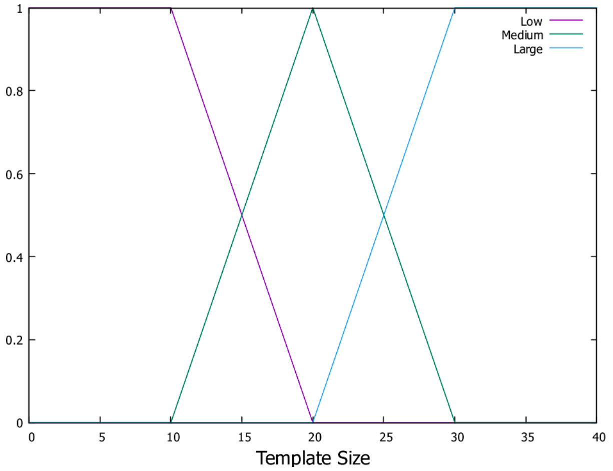


Fig. 8. Pertinence function regarding the size of the template.

Regarding the complexity of the proposed algorithm, taking into account that the number of pixels in an image corresponds to n, the proposal calculates the integral image that is directly related to the number of pixels in the image, which would result in $O(n)$ complexity. Then the operations related to the 4 templates are performed. These operations depend on the size of the template used, but as it uses regions and each operation considers two regions in the template, 8 operations are performed, so for each template applied to the entire image we would have $O(8.(n/t))$ for the four templates we would have $O(4.8.(n/t))$ which is equal to $O(32(n/t))$, where t is the template size. In the process of counting green pixels to divide by regions, the number of operations depends on the number of pixels, so the complexity would be $O(n)$. We would have the total complexity as$O(n + 32(n/t) + n)$, or simply $O(n)$.

Since Sujaritha et al. [13] is considered the most comparable to this work that we could find, we analyzed their algorithm complexity. It uses four features and a fuzzy classifier in the weed detection process, the features are: The feature extraction methods analyzed in this study include: Normalized second-order gray level co-occurrence matrix, Laws' texture features, Gabor wavelet and the rotation-robust wavelet decomposition method. Gabor Wavelet has a complexity of O(n.log(n)) [26] which is greater than the complexity of the algorithm proposed in this research. Similarly, Convolution Neural Networks, the trending on image recognition, has a complexity of O(n.log(n)) for inference, considering the lowest complexity implementation that we could find [27]. Training phase complexity varies according to the used algorithm but shouldn't be considered in a fair comparison.

Figure 9 shows an image taken in the real environment (left side), and the output image (right side). The areas were Haar-like templates matched at least two times are considered of interest for connected components. All pixels belonging to the same connected component object are marked by a red rectangle. The redder is the rectangle, the greater is the number of intersections found.

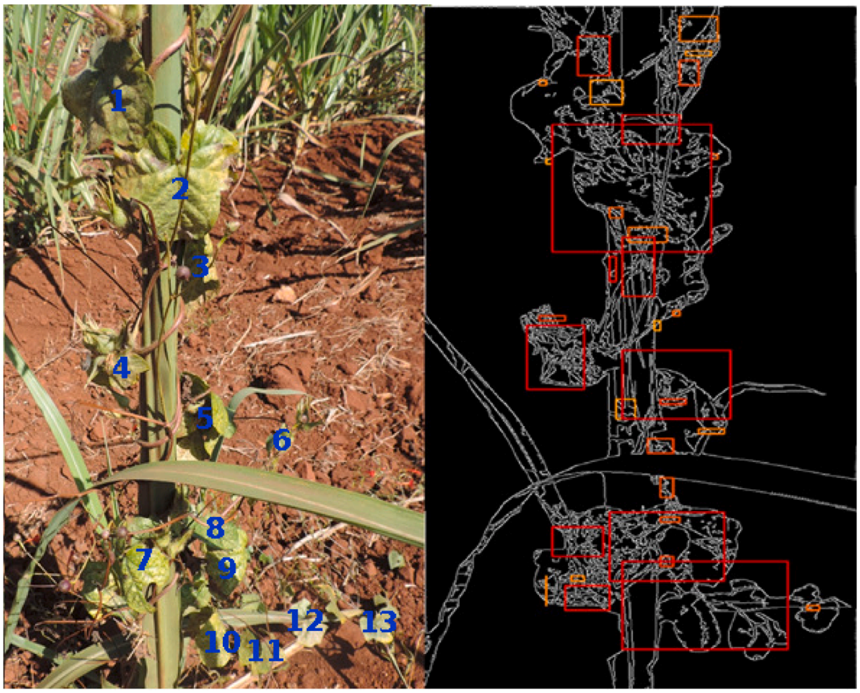


Fig. 9. Best case, only a few leaves parts are shaded.

In the best case, we would have all the leaves without shading. Figure 9 shows a scenario in which we have 13 leaves with a large visible portion. However, there are still a few shaded sections. In this case, the algorithm detected 9 of them, including leaves that are partially visible and some with leaf damage (missing parts). The accuracy obtained only with plants images with few shaded sections was 90.0%. Usually, depending on the growing stage of the plant, there will be more than one identifiable leaf, which improves the detection process.

Analyzing the images, it is possible to see that the shadow effect is the main problem in recognizing the leaves. In Fig. 10, the leaves venation are partially visible in the output image when there are shadows. In this case, the accuracy dropped to 55.5%.

Figure 11 shows the relationship between the number of shaded leaves in an image and the resulting accuracy. Images 2 and 5 have the highest number of leaves shaded, in part or total, and the lowest accuracy value. On the other hand, image 4 has the biggest accuracy value and the percentage of shaded is the lowest. Therefore, the lower the light intensity over a leaf, the greater is the difficulty to recognize it.

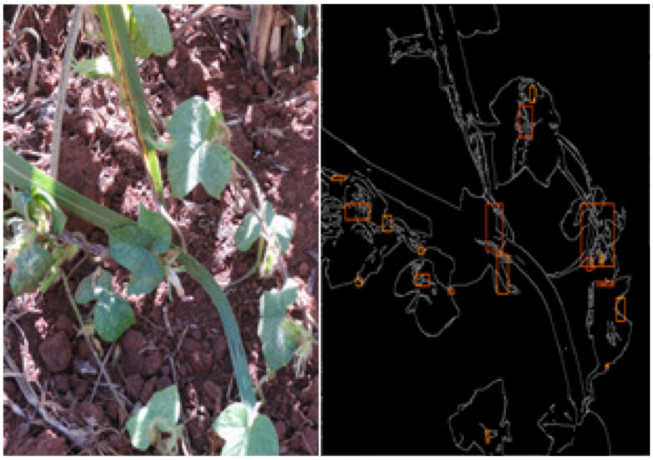


Fig. 10. One of the worst cases, most leaves with a large part shaded a) Input image from real environment (left side). b) Output image (right side).

Figures 9 and 10 shows leaves damaged by frost. Even with this kind of damage, the recognition method was successful. A similar situation is the leaf occlusion. In the real environment, it is common for the area of a leaf not to be captured in its entirety, either because it is occluded by other leaves, or by other materials such as clay, or due to the capture angle regarding leaf curvature. In this sense, we chose a leaf that was fully captured and edited the image to crop the leaf partially. Figure 12 depicts the leaf segmented as 10%, 25%, 50%, and 100% of the original image.

The identification of the connected elements that have intersections with two templates of the figure was carried out. The graphic in Fig. 13 shows the highest number of intersections obtained in each percentage of the image leaf showed in Fig. 12.

Analyzing the graph in Fig. 13, it is noticed that 50% of the leaf provides more than 20 intersections, implying a dicotyledonous leaf recognition with accuracy above 90.0%. With more than the 50% of occlusion, the number of detected intersections drops quickly, jeopardizing the dicotyledonous detection. However, our template size is proportional to the identifiable leaf part, thus our technique is invariant regarding the leaf scale on the image, providing that more than 50% of the leaf is visible.

In the tests, the number of intersections was analyzed to define a region as belonging to a dicotyledons. In the environment has other sugarcane leaves, depending on the amount of intersections threshold, there are occurrences of false negatives, intersections between sugarcane leaves defined as weeds. The number of leaves of 20 images were analyzed. The total of leaves was 86. 78 leaves were recognized, 90.7%, the 8 unrecognized leaves had less than 50% of their visible area. If a smaller number of intersections is defined, leaves with less than 50% are recognized, but sugarcane leaves are also recognized, false negatives.

5. Conclusion

The proposed method can identify *Ipomoea* spp. in a real environment, even considering the difficulties presented in such areas. The major advantage of the proposed technique is the low complexity. The Haar-like used with Fuzic Logic and Connected Components helps to reduce the space of characteristics to be investigated and provides a fast

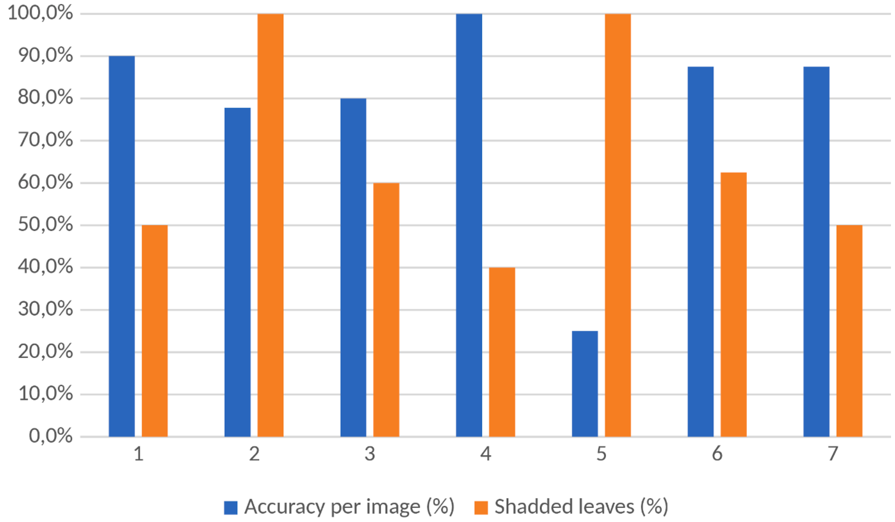


Fig. 11. The relation between accuracy and shading. The graph shows that the more shaded the leaf, the more difficult it is to classify it correctly.

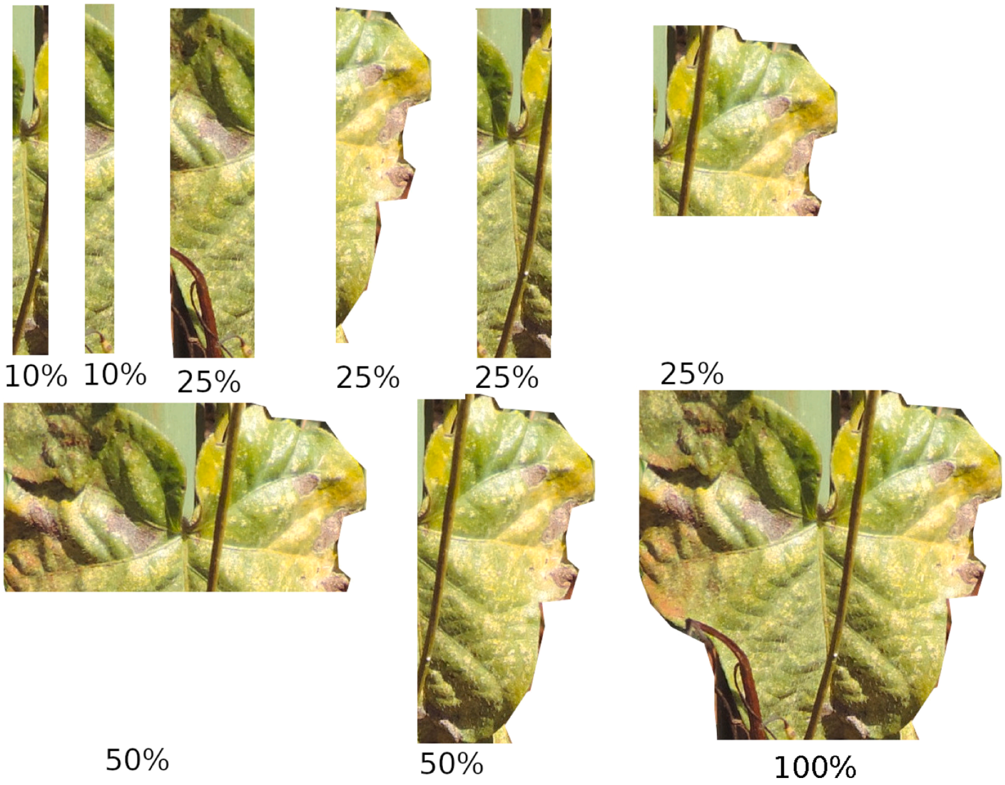


Fig. 12. Leaf image cropped to 10%, 25%, 50% and 100%.

way to identify the probable position of the leaf.

The accuracy is 90.0% in the best scenario with a low number of shaded leaves, but the light intensity was the biggest issue found in this research. The proposed technique can recognize the leaf independently of its rotations, however, for a good accuracy, the shadowed or occluded part of the leaf must be less than 50.0%.

The experiments carried out on real environment images show that the proposed method can identify leaves partially visible in the image, which shows the possibility of applying the technique to remove plants using automated artifacts such as robots. *Ipomoea* spp, at the maturity stage that demands intervention, generally have several leaves per stalk, which increases the possibility of recognition.

Future work should include a run of this technique over different times of the day and different weed maturity stages, under semi-controlled environment (with controlled light) and over different species. We also would like to run this technique over a remote embedded device with moving characteristics (such as drones or robots) to ensure the feasibility of real-time recognition and measure energy consumption.

Declaration of Competing Interest

- All authors have participated in (a) conception and design, or analysis and interpretation of the data; (b) drafting the article or revising it critically for important intellectual content; and (c) approval of the final version.

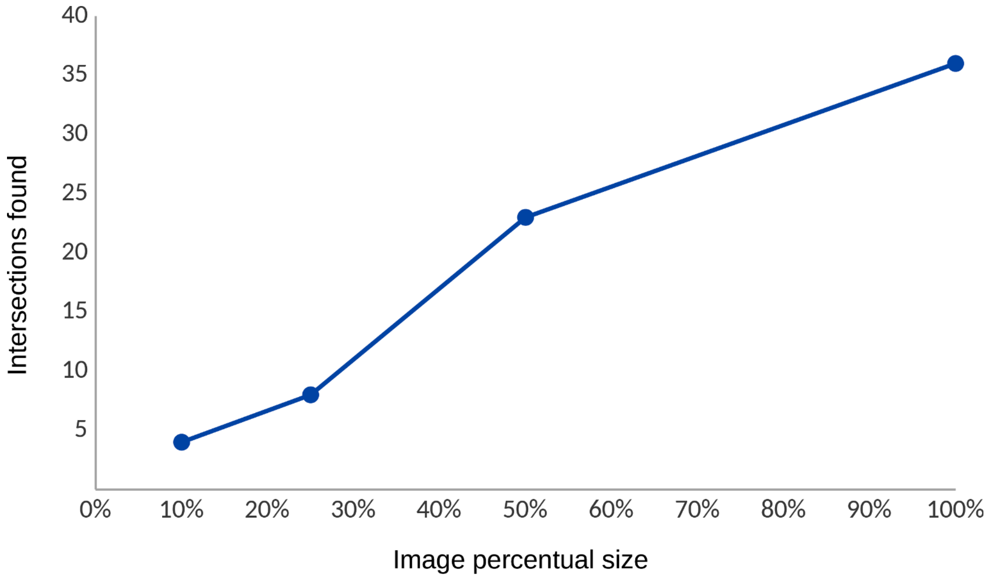


Fig. 13. Relation between leaf percentage and intersections returned.

- This manuscript has not been submitted to, nor is under review at, another journal or other publishing venue.
- The authors have no affiliation with any organization with a direct or indirect financial interest in the subject matter discussed in the manuscript
- The following authors have affiliations with organizations with direct or indirect financial interest in the subject matter discussed in the manuscript

CRediT authorship contribution statement

Josué Leal Moura Dantas: Conceptualization, Methodology, Investigation. **André Riyuiti Hirakawa:** Conceptualization, Supervision. **Bruno Albertini:** Methodology, Writing – review & editing, Supervision.

References

[1] C. Hung, J.P. Underwood, J. nd Nieto, S. Sukkarieh, Autonomous intelligent system for fruit yield estimation. Symposia on the Physiology of Perennial Fruit Crops and Production Systems and Mechanisation, Precision Horticulture and Robotics, 2016.

[2] A. Wendel, J. Underwood, Self-supervised weed detection in vegetable crops using ground based hyperspectral imaging. IEEE International Conference on Robotics and Automation (ICRA), 2016, pp. 5128–5135.Stockholm

[3] O. Bawden, F. Dayoub, J. Kulk, C. Lehnert, R. Russell, T. Perez, C. Mccool, A. English, Robot for weed species plant-specific management, Field Rob. (2017) 1179–1199.

[4] A.J. Righetto, T.G. Ramires, L.R. Nakamura, L.D. Pedro, Predicting weed invasion in a sugarcane cultivar using multispectral image, J. Appl. Stat. 46 (2019) 1–12.

[5] Z. Wang, H. Li, Y. Zhu, T. Xu, Review of plant identification based on image processing, Arch. Comput. Methods Eng. (2017) 637–654.

[6] A.L. Codizar, G. Solano, Plant leaf recognition by venation and shape using artificial neural networks. 7th International Conference on Information, Intelligence, Systems & Applications (IISA), 2016.

[7] X. Zhu, M. Zhu, H. Ren, Method of plant leaf recognition based on improved deep convolutional neural network, Cogn. Syst. Res. (2018) 223–233.

[8] Y. Cao, C. Yan, J. Li, H. Zhou, Leaf vein extraction and angle measurement using hue information and line detection. 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC) vol. 1, 2017, pp. 223–225.

[9] J.W. Tan, S. Chang, S. Abdul-kareem, H.J. Yap, K. Yong, Deep learning for plant species classification using leaf vein morphometric, IEEE/ACM Trans. Comput. Biol. Bioinf. (2020) 82–90.

[10] L. Zhang, A leaf vein detection scheme for locating individual plant leaves. International Conference on Information and Communication Technology Robotics (ICT-ROBOT), 2018.Busan

[11] C. Arrasco, S. Khlebnikov, A. Oncevay, C. Beltr, Leaf venation enhancing for texture feature extraction in a plant classification. 2018 IEEE Latin American Conference on Computational Intelligence (LA-CCI), 2018, pp. 1–4.Gudalajara, Mexico

[12] H. Kolivand, B. Mei, T. Saba, M. Shafry, M. Rahim, A. Rehman, Research article - computer engineering and computer science a new leaf venation detection technique for plant species classification, Arabian J. Sci. Eng. 44 (2019) 3315–3327.

[13] M. Sujaritha, S. Annadurai, J. Satheeshkumar, S.K. Sharan, L. Mahesh, Weed detecting robot in sugarcane fields using fuzzy real time classifier, Comput. Electron. Agric. 134 (2017) 160–171.

[14] Y. Zhao, L. Gong, Y. Huang, C. Liu, A review of key techniques of vision-based control for harvesting robot, Comput. Electron. Agric. 127 (2016) 311–323.

[15] J. Canny, A computational approach to edge detection, IEEE Trans. Pattern Anal. Mach. Intell. 8 (6) (1986) 679–698.

[16] A.M. Jagtap, V. Kangale, K. Unune, P. Gosavi, A study of LBPH, eigenface, fisherface and haar-like features for face recognition using openCV. Proceedings of the International Conference on Intelligent Sustainable Systems, 2019.ICISS, Palladam

[17] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001.

[18] Y. Li, W. Lu, S. Wang, X. Ding, Local haar-like features in edge maps for pedestrian detection. Proceedings - 4th International Congress on Image and Signal Processing vol. 3, 2011, pp. 1424–1427.CISP

[19] B. Mohamed, A. Issam, A. Mohamed, B. Abdellatif, ECG image classification in real time based on the haar-like features and artificial neural networks, Procedia Comput. Sci. (2015) 32–39.

[20] K. Peuwnuan, K. Woraratpanya, K. Pasupa, Modified adaptive thresholding using integral image. 13th International Joint Conference on Computer Science and Software Engineering, 2016.JCSSE, Khon Kaen

[21] L.F. He, Y.Y. Chao, K. Suzuki, An algorithm for connected-component labeling, hole labeling and euler number computing, J. Comput. Sci. Technol. (2015) 468–478.

[22] Z. Li, L. Li, K. Yan, C. Zhang, Automatic image annotation using fuzzy association rules and decision tree, Multimedia Syst. (2017) 679–690.

[23] B. Bataineh, S. Norul, H. Sheikh, K. Omar, Expert systems with applications a novel statistical feature extraction method for textual images: optical font recognition, Expert Syst. Appl. (2012) 5470–5477.

[24] K. Juneja, C. Rana, Multi-featured and fuzzy-filtered machine learning model for face expression classification, Wirel. Pers. Commun. (2020) 1227–1256.

[25] Q. Liu, X.P. Yang, X.L. Zhao, W.J. Ling, F.P. Lu, Y.X. Zhao, Microscopic image enhancement of chinese herbal medicine based on fuzzy set. International Conference on Image, Vision and Computing Microscopic, 2nd ed., 2017.Chengdu, China

[26] S. QIU, F. ZHOU, P.E. CRANDALL, Discrete Gabor transforms with complexity O (NlogN), Signal Process. 77 (1999) 159–170.

[27] J. PARK, J. LEE, D. SIM, Low-complexity CNN with 1D and 2D filters for super-resolution, J. Real-Time Image Process. 17 (6) (2020) 2065–2076.

Update

Smart Agricultural Technology

Volume 6, Issue , December 2023, Page

DOI: https://doi.org/10.1016/j.atech.2023.100375

Contents lists available at ScienceDirect

Smart Agricultural Technology

journal homepage: www.journals.elsevier.com/smart-agricultural-technology

Erratum to “A Deterministic Technique for Identifying Dicotyledons in Images” [ATECH Vol 3, 2022 /100092]

Josué Leal Moura Dantas [a,*], André Riyuiti Hirakawa [b], Bruno Albertini [b]

[a] *Federal Rural University of Amazonia, PA 275, km 13, Parauapebas, 68515-000, Pará, Brazil*
[b] *University of São Paulo, Av. Prof. Luciano Gualberto, tv 3, 158, São Paulo, 05508-010, São Paulo, Brazil*

The publisher regrets that the article https://doi.org/10.1016/j.atech.2022.100092 has two contradictory statements about the authors’ conflict of interest. The correct statement provided by the author is as follows:

- All authors have participated in (a) conception and design, or analysis and interpretation of the data; (b) drafting the article or revising it critically for important intellectual content; and (c) approval of the final version.
- This manuscript has not been submitted to, nor is under review at, another journal or other publishing venue.
- The following authors have affiliations with organizations with direct or indirect financial interest in the subject matter discussed in the manuscript:

Author’s name	Affiliation
Josué Leal Moura Dantas	Federal Rural University of Amazonia
Bruno Albertini	University of São Paulo
André Hirakawa (in Memoriam)	University of São Paulo

The publisher would like to apologise for any inconvenience caused.

DOI of original article: https://doi.org/10.1016/j.atech.2022.100092.

[*] Corresponding author.

E-mail address: josueleal@usp.br (J.L.M. Dantas).

https://doi.org/10.1016/j.atech.2023.100375

Available online 5 December 2023