




## Graph Cut Segmentation and Watershed Algorithm for Yield Count of an Arecanut Bunch



Anitha Arekattedoddi Chikkalingaiah<sup>1\*</sup>, Shrinivasa Naika Chikkathore Palya Laxmana<sup>2</sup>, Krishna Alabujanahalli Neelegowda<sup>3</sup>

<sup>1</sup> Department of Computer Science and Engineering, Government Sri Krishnarajendra Silver Jubilee Technological Institute, Bengaluru 560001, India

<sup>2</sup> Department of Studies in Computer Science and Engineering, University Bramhappa Devendrappa Tanvarappanavar (BDT) College of Engineering, Davanagere 577004, India

<sup>3</sup> Department of Computer Science and Engineering, SJB Institute of Technology, Bengaluru 560060, India

Corresponding Author Email: [anithakrishna2008@gmail.com](mailto:anithakrishna2008@gmail.com)

Copyright: ©2024 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/isi.290629>

### ABSTRACT

**Received:** 31 October 2023

**Revised:** 10 May 2024

**Accepted:** 19 July 2024

**Available online:** 25 December 2024

#### Keywords:

*arecanut, segmentation, SLIC, graph cut, watershed, yield*

Arecanut is one of the most significant commercial crops in Southeast Asia and plays a major role in the religious and cultural functions impacting the socio-economic life of the people. Arecanut is also used in Ayurvedic and Veterinary medicines. Arecanut is used to manufacture toothpaste, soap, tea powder, vita, and wine. Accurate segmentation of the arecanut bunch removing unwanted surrounding information helps monitoring its health, maturity, and yield. Yield estimation facilitates the farmer to plan for harvesting, storage and sale. Arecanut segmentation is complex because the color of the crop changes with the brightness quality in the outdoor field and the sharpness of the color. Another common problem in arecanut crop bunch segmentation and yield count is that of partial occlusion and overlapping of nuts. Segmentation is obtained using Simple Linear Iterative Clustering (SLIC) and graph cut algorithm. Segmentation of an arecanut bunch is achieved by first converting the picture elements into superpixels employing SLIC to lower the computational costs and the effect of noise. Graph cut produces accurate and precise segmentation considering local and global information capturing fine details and contours of objects. Watershed algorithm is used to count the arecanuts from a segmented image is presented in this paper. Segmentation resulted in 85.78% IoU and 93.15% Dice score and are better compared to benchmarks. Yield count resulted in 5.4% Mean Absolute Percentage Error (MAPE), which is very good compared to other methods.

## 1. INTRODUCTION

Agriculture is crucial to feed the population in any society and human existence depends directly on it. It plays a vital role in the economy of any country and the economy of many countries in the world depends solely on agriculture. India stands as the second largest country in the world for agricultural production and contributes to the nation's economy with almost 58% of the GDP as a traditional occupation. The economy of developing countries like India depends mainly on agriculture, but in the advanced nation, it becomes business. A large increase in food production must be attained with the reduced farming land to cover the growing population across the globe, protecting the biosphere by ensuring viable agricultural procedures [1]. A balanced farming industry assures a nation of food safety, per capita and employment. In India, farmers grow a wide range of crops. Arecanut, commonly called betel nut, is one of the main plantations in South India. India stands first in the area (47%) and in production (47%) of arecanut. India's productivity is at 1.27 tonnes/hectare and is on par with world productivity. In India, it is primarily spread in Karnataka, Meghalaya, Assam, Tamil Nadu, Kerala and West Bengal. Karnataka stands first

(62.69%) both in area and the produce.

The arecanut produced in India is mostly consumed domestically. India exported 6,663 tonnes of arecanut worth Rs. 158.26 crores in the year 2021-22, which is less than a five percent share of the global market. The major countries to which India exported arecanut were Bangladesh, UAE and Vietnam. Both production and yield of arecanut in India registered increasing trends. Arecanuts plays a vital role in sacred, communal and cultural functions impacting the socio-economic life of the people in India. The most common usage is mastication with betel leaves. Arecanut also finds its usage in Ayurvedic and Veterinary medicines. It is mainly recommended for removing tapeworms and other intestinal worms. Consumption of arecanut with betel leaf gives a natural mouth freshener and a laxative. It also helps in digestion, is a diuretic, strengthens heart muscles and regulates menstrual flow [2]. Arecanut is used to manufacture toothpaste, soap, tea powder, vita, and wine. There are many uses of the arecanut plant also. Almost every component of the tree is used by humans for one or more purposes including coloring clothes, leather tanning, and as an adhesive and safe food coloring agent. Arecanut leaf sheaths are used for preparing disposable cups, plates, tea chests, caps, packing cases, rain

protectors and suitcases.

Areca trees are very tall and grows usually to heights of 30 meters or more based on the climate, making it challenging for growth monitoring, disease identification and harvesting. Arecanut cultivation is laborious, tedious and requires regular observation for better yield. Crop yield estimation helps growers with the proper usage of resources to increase their income. Yield estimation facilitates the farmer to plan for harvesting, storage and sale. The manual process of yield estimation is labor-intensive, expensive and often inaccurate. Automated yield estimation became more useful.

This research work aims to estimate the yield of an arecanut bunch. Segmentation is an important step in any machine vision system to interpret an image with no human mediation and its achievement reflects on the attainment of the entire vision system. Image decomposition is a perceptual alignment of image pixels based on similarity, vicinity and continuity, which emulate an image regional and or global characteristics [3]. An automated, accurate and robust segmentation technique is required as hand-operated segmentation is hard, laborious, is subjective and often prone to error. Segmentation is challenging because of variations in the crop color, silhouette and inter-reflection in outdoor as the daylight changes. Accurate segmentation of the arecanut bunch removing unwanted surrounding information is essential to find its health, maturity, and yield. Many surveys exist on crop segmentation and yield estimation for crops like grapes, apples, mangoes, almonds, and tomatoes. However, more work is needed for the segmentation and yield estimation of arecanut.

The remainder of the article is arranged as follows. Review of the existing works related to segmentation and yield estimation in precision agriculture is presented in section 2. The methodology for arecanut bunch segmentation and yield count is detailed in section 3. Section 4 describes the results and performance details. The last division summarizes the work done and the future avenues.

## 2. LITERATURE SURVEY

A review of existing work on segmentation and yield estimation of crops like grapes, apples, mangoes, almonds, and tomatoes is directly related to segmentation and yield estimation of arecanut since the images are captured in the outdoor field with natural illumination containing complex backgrounds with occlusion and overlapping. Most of the yield assessment methods require object segmentation/detection. Most of the existing segmentation approaches concentrate on a two-class classification method, i.e. object and the background. Background elimination is the first phase and must be accomplished in a more pertinent manner to avoid misclassification. Rahman and Hellicar [4] proposed a method to segment mature grape bunch by first finding the edges using gradients of brightness images and then identifying circles using circular Hough transform. These circles are classified as grapes or backgrounds using a learned classifier. As the maximum count of grapes tends to remain in spatial proximity to each other, circles with no neighbor within two times the size of its diameter are considered detached and are removed. So, the rest of the circles are sorted into groups depending on their proximity using k-means clustering.

Berenstein et al. [5] proposed three grape detection algorithms. The first one depends on the distinction in edge disposal among the foliage and grape cluster. Foliage regions

contain less edges than those in grape clusters. The second one is using the decision tree algorithm C5.0. Images are stored both in RGB and HSV representation. The patches from grape and foliage areas are utilized to extract the mean, standard deviation of the gradient magnitude from each R, G, B, H, S and V channel. The third one depends upon comparison of pixels between edge representations of the input image and an edge filter that represents grapes.

Font et al. [6] compared the performance of different pixel-based approaches to identify reddish grapes: Color-based segmentation using RGB and HSV, Otsu threshold-based segmentation applied to R, G, B, I and H layers, Mahalanobis distance-based segmentation using the variance of each intensity layer accomplished by comparing the templates of the grape and the background, Bayesian classifier-based segmentation with simplified discriminant analysis using covariance matrices of the templates that characterize the grapes and the background, Linear color model-based segmentation choosing a part of object region in an image whose pixel intensity has a linear relationship that can be modelled with linear regression. Histogram-based segmentation dilate color correlation of the 3-D color histogram obtained from the template of grape, convolve the histogram along with a solid sphere and then segment using a zero-threshold applied to the histogram. Threshold-based segmentation of the H layer gave a better result for nonoccluded reddish grapes.

Initially, Guru and Shivamurthy [7] proposed a framework to segment mango regions applying adaptive thresholding to each of the color bands R, G and B individually and are merged back later. Smoothing and binarization of the merged image give the position of mangoes. Texture features derived for each position are then compared with templates kept in the repository to remove noisy regions. Edges are extracted from the regions located and superimposed them onto the localized image for segmentation to correctly depict the boundaries of all the mangoes identified and clearly differentiate mangoes in case of occlusion. A method for estimating mango crop yield based on texture and color features proposed by Payne et al. [8] using the normalized difference index determined for each pixel from R and G layers. RGB picture was then processed with a  $3 \times 3$  variance filter to replace every pixel by the neighborhood variance of R, G and B layers, correspondingly. The resultant image has been transformed to grayscale and thresholded. A binary image was then produced by aggregating the outputs of the preceding stages, effectively masking the mango regions. Lastly, determine the number of fragments in the binary image using lower and upper limits on the count of pixels in the particle considered.

Verma et al. [9] proposed a framework for segmenting tomatoes based on active contour. Since tomato position changes as the season progresses, they proposed to approximate the movement among the two successive images by computing SIFT descriptors at contour points. Gradient information is then fused with region information to introduce an elliptic approximation of the tomato boundary. Four elliptical estimates have been determined using Active contour with an elliptical shape constraint, out of which the superior one has chosen as the final segmentation. Hung et al. [10] proposed a framework to tackle the multiple-class (crop, leaves and branches) issue applied to almond. This approach uses feature learning with a Conditional Random Field (CRF) to determine the rule set automatically from the data itself, an alternative to the fixed predefined feature descriptors. This

approach follows unsupervised learning to derive the most relevant attributes of the data automatically. This method is more suitable for various crops because it can handle inherent variance. It automatically adapts the feature sets for dissimilar crops. So, the method will not require domain-dependent assumptions and may be applied to variety of crops.

Wang et al. [11] suggested a system for yield estimation of apple orchards using hue, saturation and value features for red apple detection and hue, saturation and intensity profiles to detect green apple pixels. Morphological methods were then used to convert apple regions into apple counts. Bargoti and Underwood [12] proposed a structure for segmentation and yield estimation in apple orchards using general-purpose feature learning methods: Multi-scale Multi-Layered Perceptrons (MLP) and Convolutional Neural Networks (CNN). The architectures were expanded by adding contextual information regarding how the image data has acquired (metadata) to find the relationships between meta-parameters and the object classes.

Watershed segmentation and circular Hough transform algorithms have applied to pixel-wise segmented results to detect and count individual fruits. Li et al. [13] proposed in-field cotton identification using simple linear iterative clustering (SLIC) and DBSCAN with Wasserstein distance to produce regions, followed by semantic classification using random forest with the derivation of histogram-based color and texture features of regions.

Siddesha et al. [14] explored different techniques to segment arecanut bunch. Threshold-based segmentation neglects spatial information, and the choice of threshold value is critical as it may lead to over or under-segmentation. For k-means clustering, desired count of clusters needs to be specified. There needs to be more clarity in selection of features for good results in fuzzy C-means clustering. Histograms may enhance the speed for fast fuzzy C-means clustering. Watershed segmentation suffers from over-segmentation. It gives connected components at the cost of computational time. Maximum similarity-based region merging is computationally slow as it needs human intervention but gives better results than other techniques.

Segmentation of arecanut bunch using active contouring by converting the input image into grayscale, then to binary followed by morphological operations, active contouring, and finally converting back to RGB presented by Dhanesha and Shrinivasa Naika [15]. It does the segmentation using a mask. Initially, the mask size equals the input image size and then decreases as the object's contour. Here, the observation made is experimentation done with only 20 images. Further, they have continued and proposed segmentation by first converting the RGB into YCbCr model and then converting it into binary using threshold values 120 and 200, followed by erosion and closing using 4-neighbourhood connectivity. The result achieved is 80% for 500 images, including immature, mature and over-matured images [16]. Authors further extended the work and analyzed the accuracy of various color models: YUV, YCbCr, YCgCr, YPbPr and HSV [17]. They have also increased the database size to 1017 and demonstrated that YCgCr and HSV color models are better than others.

Areca bunch is a natural cluster of arecanuts similar to a grape bunch that grows on a branch coming out of the stem with color variation during the growth. A large number of nuts are in the spatial vicinity of each other similar to grape clusters. We translated these cues into the segmentation and yield count. Segmentation and yield estimation are done as an off-line task

for most of the crops. The methods proposed are specific to a crop. The survey concluded that more research is required to extract arecanut bunch for disease detection, classification, identification of maturity levels, and yield estimation.

### 3. METHODOLOGY

In this division, we discuss the methodology followed to segment the arecanut bunch and estimate the yield. The flow of the work done is depicted in Figure 1. Segmentation is achieved by first converting the picture elements into superpixels using simple linear iterative clustering (SLIC) to reduce computational costs and the effects of noise. The user must interactively mark the object and the background on the superpixel image as seed points [18]. A graph is then constructed using the superpixels as nodes and the marked regions as two end vertices. Graph cut is employed to obtain a mask image using min-cut/max-flow algorithm to solve the energy function. Segmented image is obtained by performing pixel multiplication of the mask and the input image. The segmented image is then fed to the watershed algorithm to calculate the yield count of arecanut in the image.

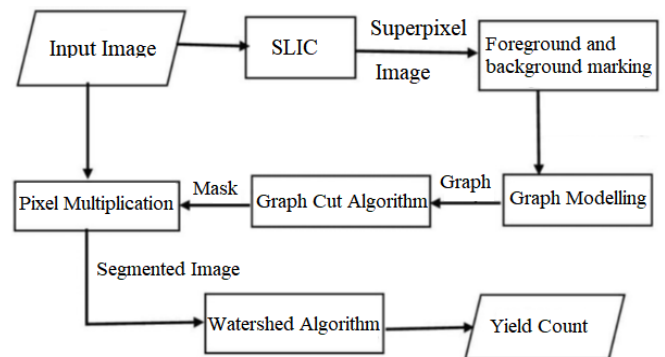


Figure 1. Architecture of arecanut bunch segmentation and yield count

#### 3.1 Converting the image into superpixels using SLIC

SLIC addresses two inherent problems of digital image processing, namely: discretization and computational complexity. SLIC groups pixels which are perceptually similar and decrease the count of primitives for subsequent algorithms, thereby improving the accuracy and efficiency of image segmentation [19]. SLIC is among the easiest and most vigorous methods to group the homogeneous regions of an image, and it adopts a new distance measure taking full advantage of color and spatial information to enhance the shape of superpixels. SLIC controls the count of superpixels using only one parameter, and it produces superpixels by grouping pixels based on the similarity of color and distance.

SLIC uses five-dimensional space  $[labxy]$  in CIELAB color space, where  $[lab]$  is color vector of the picture element and  $[xy]$  is the pixel coordinate. So, we have to convert the given image into CIELAB color space. The input for SLIC is  $K$ , the required number of superpixels almost equal in size. Initialize all the  $K$  superpixel cluster centres  $C_k = [l_k, a_k, b_k, x_k, y_k]$  at uniform grid intervals,  $S = \sqrt{N/K}$ ,  $N$  be the count of image elements in the input image. The spatial extent of any superpixel is roughly  $S^2$ . The normalized distance  $D_s$  used in

5D space is the sum of  $d_{lab}$  distance and  $d_{xy}$ ,  $xy$  plane distance given in Eq. (1) [20]. Parameter  $m$  is utilised to manage the size of a superpixel. A higher value results in a more compact cluster and is determined to be within the range [1, 20].

$$D_s = d_{lab} + (m/S) * d_{xy} \quad (1)$$

where,

$$d_{lab} = \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2},$$

$$d_{xy} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}.$$

SLIC begins by instantiating  $K$  cluster centres, which are scattered evenly and make them seed points that resemble the lowest gradient point in a  $3 \times 3$  neighbourhood. Image gradients are computed using Eq. (2). Each picture element is assigned to the nearby cluster centre, and then a new centre is determined by averaging all the pixels of clusters for the [labxy] vector. The method of assigning pixels to the closest cluster centre and recalculating the cluster centre is iterated until convergence. A few drift labels may exist at the end of this operation. i.e., pixels in the neighbourhood of a more prominent segment but not related to it. The concluding phase of the method must ensure connectivity by relabelling disjunct segments with the most prominent neighbouring cluster label.

$$G(x, y) = ||I(x + 1, y) - I(x - 1, y)||^2 + ||I(x, y + 1) - I(x, y - 1)||^2 \quad (2)$$

where,  $I(x, y)$ , the lab vector representing image element at  $(x, y)$  position and  $||\cdot||$  represents  $L2$  norm.

### 3.2 Interactive segmentation using graph cut

The fundamental idea of interactive graph cut segmentation is that the user marks the object and the surroundings, indicating that specific pixels (seeds) are a portion of the foreground and specific pixels are a portion of the background. This indication provides a hint about what the end user aspires to segment. Then an image is automatically segmented by determining a global optimal cut of all segmentations. Image segmentation is considered as a labelling problem in a graph. Image segmentation allocates labels to each element in an image so that elements with the same label share certain features or properties. Every pixel in a specific region must be similar concerning some attribute or property, such as color, intensity, intensity or texture.

In general, an image is represented as a vector of pixels. This work modifies pixel-based representation to superpixel-based representation. An input image is represented as a vector of the form  $I = (I_1, I_2, \dots, I_i, \dots, I_n)$ , while  $I_i$  represents color vector of the superpixels  $i$ . Each superpixel will be treated as a node, and the undirect graph  $G = (V, E)$  will be formed by connecting these nodes [21]. Here,  $V$  serve as the set of all nodes, and  $E$  serve as the difference between the histograms of the two neighbouring superpixels. All the nodes are connected to two endpoints, namely: source  $S$  and terminal  $T$  vertices.

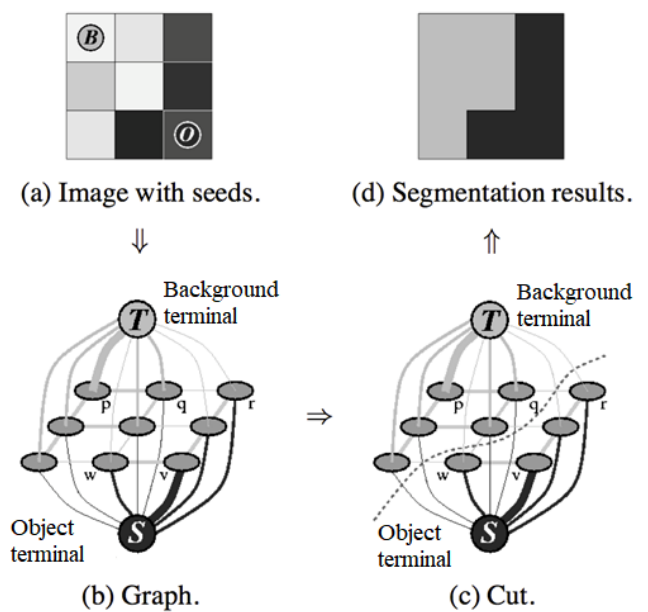
Figure 2(a) displays a case of  $3 \times 3$  image with user markings of the object and the background. The corresponding graph with two terminals  $S$  and  $T$  is shown in Figure 2(b). All the pixels except the object are joined to  $T$ , the background terminal, and all the pixels except the background are attached to  $S$ , the object terminal. The thickness of the edges shows edge costs. Next, calculate the global minimal cut (Figure 2(c)),

splitting these two terminals and giving a segmentation result (Figure 2(d)) which precisely represents an "object" and "background" region [22].

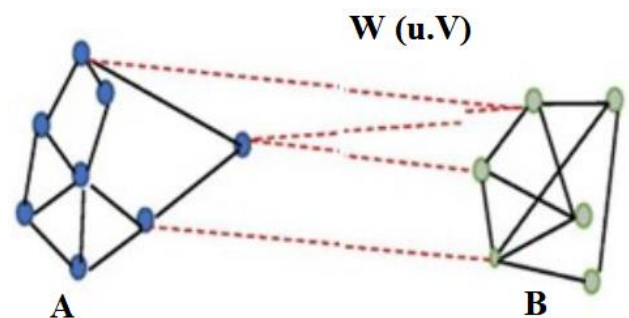
In general, graph-based image segmentation divides  $G$  into mutually exclusive non-empty sets  $A_1, A_2, \dots, A_k$  such that  $A_i \cap A_j = \emptyset$  ( $i, j \in \{1, 2, \dots, k\}, i \neq j$ ) and  $A_1 \cup \dots \cup A_k = G$ . The criteria for segmentation are that elements within a group are similar and dissimilar across the group. The dissimilarity between two components,  $A$  and  $B$ , is considered graph cuts. A cut is a subgroup of edges that splits the graph into two separate sets  $A$  and  $B$  and the value of cut is determined as the sum of edge weights removed between these two sets and is given in Eq. (3).

$$cut(A, B) = \sum_{u \in A, v \in B} w(u, v) \quad (3)$$

where,  $u$  and  $v$  are the vertices in two distinct components.



**Figure 2.** A simple example for segmentation of  $3 \times 3$  image: B - background, O - object. Edge cost is reflected by its thickness



**Figure 3.** Cut of a graph

An S-T cut in two-class segmentation is a subset  $C \subseteq E$  of edges such that it separates terminals  $S$  and  $T$ , shown in Figure 2(c). The objective is to determine the best cut for an "optimal" segmentation [23]. In graph cut, a weighted graph is split into disjoint sets (groups) such that similarity within a group is high and that across the group is low. Figure 3 shows the level of

dissimilarity between the two groups  $A$  and  $B$  determined as the sum of weights of edges removed between these two pieces. One can perfectly bipartition the graph and accomplish superior segmentation by lessen this cut value. Normalized cut given in Eq. (4) calculates the cut cost as the ratio of the sum of the edge weights removed among these two sets and the complete edge links to all the vertices in a graph [24].

$$N_{cut} = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)} \quad (4)$$

where,  $assoc(A, V) = \sum_{u \in A, t \in V} w(u, t)$  is a total connection from the vertices of set  $A$  to all the vertices in the graph. The output of the graph cut is the mask image. The final segmented image is obtained by pixel multiplication of the masked image with the input image. This segmentation output is be used as an input for yield count. To get the yield count, we have used the watershed algorithm.

### 3.3 Yield count using watershed algorithm

Watershed is a classic segmentation algorithm and is very much useful when extracting overlapping objects in an image. The main steps of yield count are as follows [25]:

**Step 1:** The first step is to convert the segmented image into gray scale by which we can visualize the picture as a topographic surface in which low-intensity denotes valleys and high-intensity values denote peaks and hills.

**Step 2:** Otsu thresholding is employed to segment the foreground image from the background.

**Step 3:** We now apply a watershed algorithm to count the arecanuts:

- i. The first step is determining the Euclidean distance to the background pixel for every foreground pixel and then discovering local maxima (i.e., peaks). The distance between peaks is at least 34 pixels.
- ii. We then apply an 8-connected component analysis to obtain markers, fed to a watershed algorithm that assumes markers constitute valleys (i.e., local minima) in the distance map, the negative value of the distance map.
- iii. The output of the watershed algorithm is a matrix of labels. Pixels that have the same label are considered as one arecanut.
- iv. The final step is to loop over unique labels to extract each arecanuts simply.

## 4. PERFORMANCE ANALYSIS

The data set made available by Dhanesha et al. [16] has been used to assess the performance of segmentation and yield count. The data set consists of 1017 images, including unripe (629) and ripe (388) images. All the images are of 4160x3120 resolution saved in jpeg format. Segmentation performance has been assessed using Intersection-over-Union (IoU)/Jaccard index and Dice similarity coefficient shown in Eqs. (5) and (6), respectively. IoU is determined as the degree of overlap between ground truth and prediction region. Dice is determined as two times the number of pixels common to two sets divided by the sum of the pixels in each set. There are very few efforts made to extract arecanut bunch which are based on color models [16, 17] and using deep learning techniques [26, 27]. Table 1 summarizes the segmentation performance and analogy with state-of-the-art methods. The higher values of the

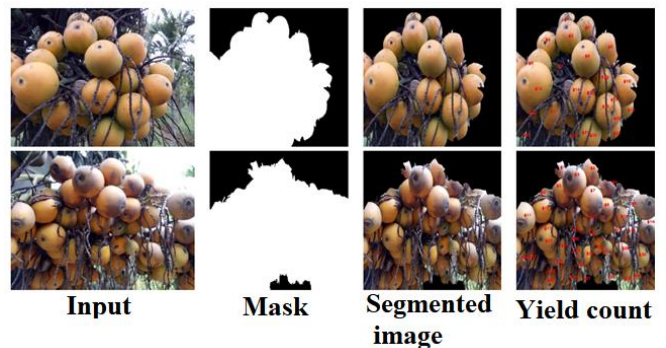
above measures indicate the greater performances of the segmentation algorithm. Results show that the graph cut segmentation model illustrated greater segmentation performance and was robust to handle variations such as scale, background, illumination, inflorescence and color.

$$IoU = \frac{|A \cap B|}{|A \cup B|} \quad (5)$$

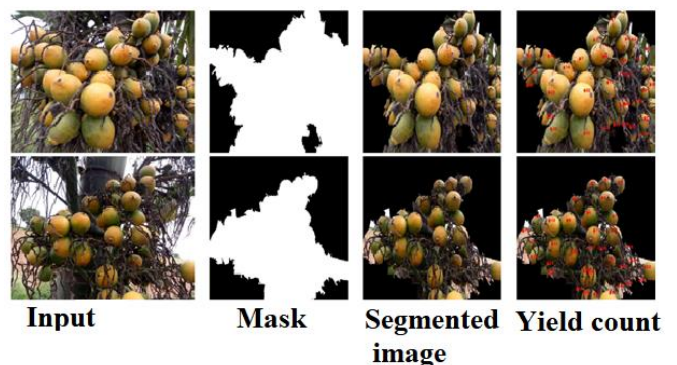
$$Dice = 2 \cdot \frac{|A \cap B|}{|A \cup B|} \quad (6)$$

**Table 1.** Comparison of segmentation performance

Author	Method	IoU	Dice
Dhanesha et al. [16]	YCgCr		80.96%
Dhanesha et al. [17]	HSV	66.58%	79.0%
	YCgCr	72.77%	83.62%
Anitha et al. [26]	U-Net	56.34%	70.61%
	MRCNN	63.49%	75.81%
Chikkalingaiah et al. [27]	U2-Net	68.49%	81.27%
	<b>SLIC, Graph-cut</b>	<b>85.78%</b>	<b>93.15%</b>



**Figure 4.** Representative results for ripe images



**Figure 5.** Representative results for unripe images

Using the watershed algorithm, we can identify and count arecanut that overlap or touch each other. Yield count performance has been determined using Mean Absolute Percentage Error (MAPE) given by (7) on counting arecanuts for 20 images. MAPE has been determined to be 5.34% for ripe images (Table 2) and 5.47% for unripe images (Table 3). The model performance is slightly better for ripe images compared to unripe images. The model performance is compared with methods applied to other crops as there were no benchmarks and is better compared to apple crop load

estimation [28, 29] with 21.1% and 15% of MAPE, respectively. The sample segmentation and yield outputs achieved are presented for both ripe and unripe images in Figure 4 and Figure 5, respectively.

$$MAPE = \frac{1}{N} \sum_i^N \frac{|n_a - n_b|}{n_a} \quad (7)$$

where,

- $n_a$ : Actual number of arecanuts in a bunch
- $n_b$ : Number of arecanuts detected by the algorithm
- $N$ : Number of sample images considered

**Table 2.** Yield count error for segmented ripe images

Image	Actual	Detected	Error (%) $ n_a - n_b $
1	46	44	4.34
2	26	26	0
3	35	38	8.57
4	42	39	7.14
5	21	25	19.04
6	25	24	4
7	31	34	9.67
8	41	39	4.87
9	34	33	2.94
10	14	14	0
11	40	41	2.5
12	33	33	0
13	45	44	2.2
14	46	42	4.34
15	26	26	0
16	70	68	2.85
17	32	36	12.5
18	24	26	8.33
19	35	32	8.57
20	40	42	5.0
MEPE			5.34

**Table 3.** Yield count error for segmented unripe images

Image	Actual	Detected	Error (%) $ n_a - n_b $
1	36	36	0
2	42	46	9.52
3	49	47	4.08
4	35	34	2.85
5	35	35	0
6	44	41	6.81
7	38	40	5.26
8	38	37	2.63
9	44	40	9.09
10	32	29	9.37
11	38	36	5.26
12	21	23	9.52
13	42	42	0
14	44	43	2.27
15	48	43	10.41
16	21	20	4.76
17	22	24	9.09
18	44	45	2.27
19	24	22	8.33
20	38	41	7.89
MAPE			5.47

## 5. CONCLUSIONS

SLIC and graph cut techniques are used to segment the

arecanut bunch and watershed algorithm to count the arecanuts in a bunch presented in this paper. Results show that graph cut model illustrated greater segmentation performance resulted in 85.78% IoU and 93.15% Dice score. Yield count using watershed also demonstrated very good performance resulted in 5.4% MAPE, which is very good in contrast with the state-of-the-art methods. Both segmentation and yield count methods are robust to handle variations such as scale, background, illumination, inflorescence and color. Here, the challenges are more due to inflorescence and occlusion. The performance could be improved by removing the inflorescence (male flowers) and automating the selection of seed points for segmentation. Segmentation and yield estimation are done as an off-line task. They may be extended as an on-line work. The methods implemented is tested for arecanut. It may further extend to generalize for different crops.

## REFERENCES

- [1] McCabe, M.F., Houborg, R., Lucieer, A. (2016). High-resolution sensing for precision agriculture: From Earth-observing satellites to unmanned aerial vehicles. In Remote Sensing for Agriculture, Ecosystems, and Hydrology XVIII, 9998: 346-355. <https://doi.org/10.1117/12.2241289>
- [2] Uses of areca and its components. <https://www.tssindia.in/index.php/other1/about-arecanut-menu/uses-of-areca-and-components>.
- [3] Peng, B., Zhang, L., Zhang, D. (2013). A survey of graph theoretical approaches to image segmentation. Pattern Recognition, 46(3): 1020-1038. <https://doi.org/10.1016/j.patcog.2012.09.015>
- [4] Rahman, A., Hellicar, A. (2014). Identification of mature grape bunches using image processing and computational intelligence methods. In 2014 IEEE Symposium on Computational Intelligence for Multimedia, Signal and Vision Processing (CIMSIVP), Orlando, FL, USA, pp. 1-6. <https://doi.org/10.1109/CIMSIVP.2014.7013272>
- [5] Berenstein, R., Shahar, O.B., Shapiro, A., Edan, Y. (2010). Grape clusters and foliage detection algorithms for autonomous selective vineyard sprayer. Intelligent Service Robotics, 3: 233-243. <https://doi.org/10.1007/s11370-010-0078-z>
- [6] Font, D., Tresanchez, M., Martínez, D., Moreno, J., Clotet, E., Palacín, J. (2015). Vineyard yield estimation based on the analysis of high-resolution images obtained with artificial illumination at night. Sensors, 15(4): 8284-8301. <https://doi.org/10.3390/s150408284>
- [7] Guru, D.S., Shivamurthy, H.G. (2013). Segmentation of mango region from mango tree image. In Mining Intelligence and Knowledge Exploration: First International Conference, MIKE 2013, Tamil Nadu, India, pp. 201-211. [https://doi.org/10.1007/978-3-319-03844-5\\_21](https://doi.org/10.1007/978-3-319-03844-5_21)
- [8] Payne, A.B., Walsh, K.B., Subedi, P.P., Jarvis, D. (2013). Estimation of mango crop yield using image analysis-segmentation method. Computers and Electronics in Agriculture, 91: 57-64. <https://doi.org/10.1016/j.compag.2012.11.009>
- [9] Verma, U., Rossant, F., Bloch, I. (2015). Segmentation and size estimation of tomatoes from sequences of paired images. EURASIP Journal on Image and Video

- Processing, 2015: 1-23. <https://doi.org/10.1186/s13640-015-0087-0>
- [10] Hung, C., Nieto, J., Taylor, Z., Underwood, J., Sukkarieh, S. (2013). Orchard fruit segmentation using multi-spectral feature learning. In 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, pp. 5314-5320. <https://doi.org/10.1109/IROS.2013.6697125>
- [11] Wang, Q., Nuske, S., Bergerman, M., Singh, S. (2013). Automated crop yield estimation for apple orchards. In Experimental robotics: The 13th International Symposium on Experimental Robotics, Springer International Publishing, pp. 745-758. [https://doi.org/10.1007/978-3-319-00065-7\\_50](https://doi.org/10.1007/978-3-319-00065-7_50)
- [12] Bargoti, S., Underwood, J.P. (2017). Image segmentation for fruit detection and yield estimation in apple orchards. *Journal of Field Robotics*, 34(6): 1039-1060. <https://doi.org/10.1002/rob.21699>
- [13] Li, Y., Cao, Z., Lu, H., Xiao, Y., Zhu, Y., Cremers, A.B. (2016). In-field cotton detection via region-based semantic image segmentation. *Computers and Electronics in Agriculture*, 127: 475-486. <https://doi.org/10.1016/j.compag.2016.07.006>
- [14] Siddesha, S., Niranjana, S.K., Aradhya, V.M. (2020). A study of different color segmentation techniques for crop bunch in arecanut. In Environmental and Agricultural Informatics: Concepts, Methodologies, Tools, and Applications, pp. 1078-1105. <https://doi.org/10.4018/978-1-5225-9621-9.ch048>
- [15] Dhanesha, R., Shrinivasa Naika, C.L. (2019). A novel approach for segmentation of arecanut bunches using active contouring. *Integrated Intelligent Computing, Communication and Security*, 771: 677-682. [https://doi.org/10.1007/978-981-10-8797-4\\_69](https://doi.org/10.1007/978-981-10-8797-4_69)
- [16] Dhanesha, R., Naika, C.S., Kantharaj, Y. (2019). Segmentation of arecanut bunches using YCgCr color model. In 2019 1st International Conference on Advances in Information Technology (ICAIT), Chikmagalur, India, pp. 50-53. <https://doi.org/10.1109/ICAIT47043.2019.8987431>
- [17] Dhanesha, R., Umesh, D.K., Naika, C.S., Girish, G. N. (2021). Segmentation of Arecanut bunches: A comparative study of different color models. In 2021 IEEE Mysore Sub Section International Conference (MysuruCon), Hassan, India, pp. 752-758. <https://doi.org/10.1109/MysuruCon52639.2021.9641680>
- [18] Li, W.L. (2013). Interactive clothing image segmentation based on superpixels and graph cuts. In 2013 International Conference on Computer Sciences and Applications, Wuhan, China, pp. 659-662. <https://doi.org/10.1109/CSA.2013.160>
- [19] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S. (2012). SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11): 2274-2282. <https://doi.org/10.1109/TPAMI.2012.120>
- [20] Xiao, F., Sun, H. (2019). Analysis and design of image segmentation algorithm based on super-pixel and graph cut. *International Journal of Advanced Network, Monitoring and Controls*, 3(4): 25-30. <https://doi.org/10.21307/ijanmc-2019-017>
- [21] Felzenszwalb, P.F., Huttenlocher, D.P. (2004). Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59: 167-181. <https://doi.org/10.1023/B:VISI.0000022288.19776.77>
- [22] Boykov, Y.Y., Jolly, M.P. (2001). Interactive graph cuts for optimal boundary & region segmentation of objects in ND images. In Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, Vancouver, BC, Canada, 1: 105-112. <https://doi.org/10.1109/ICCV.2001.937505>
- [23] Boykov, Y., Funka-Lea, G. (2006). Graph cuts and efficient ND image segmentation. *International Journal of Computer Vision*, 70(2): 109-131. <https://doi.org/10.1007/s11263-006-7934-5>
- [24] Shi, J., Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8): 888-905. <https://doi.org/10.1109/34.868688>
- [25] Rosebrock. (2015). Watershed OpenCV. <https://pyimagesearch.com/2015/11/02/watershed-opencv/>.
- [26] Anitha, A.C., Dhanesha, R., CL, S.N., Krishna, A.N., Kumar, P.S., Sharma, P.P. (2022). Arecanut bunch segmentation using deep learning techniques. *International Journal of Circuits, Systems and Signal Processing*, 16: 1064-1073. <https://doi.org/10.46300/9106.2022.16.129>
- [27] Chikkalingaiah, A.A., Dhanesha, R., Palya Laxma, S.N.C., Neelegowda, K.A., Puttaswamy, A.M., Ayengar, P. (2024). Segmentation and yield count of an arecanut bunch using deep learning techniques. *IAES International Journal of Artificial Intelligence (IJ-AI)*, 13(1): 542-553. <http://doi.org/10.11591/ijai.v13.i1.pp542-553>
- [28] Gongal, A., Silwal, A., Amatya, S., Karkee, M., Zhang, Q., Lewis, K. (2016). Apple crop-load estimation with over-the-row machine vision system. *Computers and Electronics in Agriculture*, 120: 26-35. <https://doi.org/10.1016/j.compag.2015.10.022>
- [29] Cohen, O., Linker, R., Naor, A. (2011). Estimation of the number of apples in color images recorded in orchards. In Computer and Computing Technologies in Agriculture IV: 4th IFIP TC 12 Conference, CCTA 2010, Nanchang, China, pp. 630-642. [https://doi.org/10.1007/978-3-642-18333-1\\_77](https://doi.org/10.1007/978-3-642-18333-1_77)