Robotic Aubergine Harvesting Using Dual-Arm Manipulation

DELIA SEPÚLVEDA [ID], **ROEMI FERNÁNDEZ** [ID], **EDUARDO NAVAS** [ID],
MANUEL ARMADA [ID]**, (Member, IEEE), AND PABLO GONZÁLEZ-DE-SANTOS** [ID]
Centre for Automation and Robotics CSIC-UPM, 28500 Madrid, Spain

Corresponding authors: Delia Sepúlveda (delia.sepulveda@csic.es) and Roemi Fernández (roemi.fernandez@csic.es)

ABSTRACT Interest in agricultural automation has increased considerably in recent decades due to benefits such as improving productivity or reducing the labor force. However, there are some current problems associated with unstructured environments make developing a robotic harvester a challenge. This article presents a dual-arm aubergine harvesting robot consisting of two robotic arms configured in an anthropomorphic manner to optimize the dual workspace. To detect and locate the aubergines automatically, we implemented an algorithm based on a support vector machine (SVM) classifier and designed a planning algorithm for scheduling efficient fruit harvesting that coordinates the two arms throughout the harvesting process. Finally, we propose a novel algorithm for dealing with occlusions using the capabilities of the dual-arm robot for coordinate work. Therefore, the main contribution of this study is the implementation and validation of a dual-arm harvesting robot with planning and control algorithms, which, depending on the locations of the fruits and the configuration of the arms, enables the following: (i) the simultaneous harvesting of two aubergines; (ii) the harvesting of a single aubergine with a single arm; or (iii) a collaborative behavior between the arms to solve occlusions. This cooperative operation mimics complex human harvesting motions such as using one arm to push leaves aside while the other arm picks the fruit. The performance of the proposed harvester is evaluated through laboratory tests that simulate the most common real-world scenarios. The results show that the robotic harvester has a success rate of 91.67% and an average cycle time of 26 s/fruit.

INDEX TERMS Aubergines, dual-arm robot manipulation, eggplants, intelligent perception, machine vision, occlusions, precision agriculture, robotic harvesting.

I. INTRODUCTION

In recent decades, there has been a growing interest in automating the harvesting of fruits and vegetables. This interest stems from the benefits that advanced agricultural automation can provide. Robotic harvesting can improve productivity many-fold by reducing manual labor and production costs, increasing yield and quality, and enabling better control over environmental implications. However, the complexity of agricultural environments combined with the intensity of production demands requires robust systems capable of adapting to high crop variability. Two critical aspects for achieving a

The associate editor coordinating the review of this manuscript and approving it for publication was Chenguang Yang [ID].

successful automation of harvesting tasks are detecting fruits and vegetables in natural conditions and the proper grasping and manipulation of the detected target products.

There are countless challenges associated with the ability to process, analyze and interpret visual inputs in unstructured environments. In agricultural settings, scenes exhibit a large degree of uncertainty; they contain objects with various colors, shapes, sizes, textures, and reflectance properties that change continuously due to illumination and shadow conditions [1], [2]. A broad overview of the development of vision technology applied in precision agriculture applications was compiled by [2]–[5]. Severe occlusion of fruits or vegetables, which may be partially shadowed by other fruits, stems and leaves, is another common problem in real-world scenarios.

Several strategies have been proposed to address these occlusions. One popular method is the circular Hough transform, which is effective for round objects such as oranges, apples and tomatoes [6]. However, the results show that this method is not only prone to false positives produced by the contours of other objects such as leaves, but is also computationally time-consuming, which makes real-time applications challenging. Another strategy proposed the use of an air-blowing device to avoid leaf occlusion and move adjacent fruits aside [7]. However, this solution increases the weight of the end-effector and may not be applicable to all types of crops.

After the 3D position of the fruit to be harvested has been obtained, its coordinates can be further utilized to instruct the movement of a robotic arm. Numerous harvesting robots based on this approach have been proposed in the literature for different kinds of crops [8], [9]. In [9], a 4 DoF manipulator guided by a 3-D vision system was proposed for picking cherries, while [8] proposed a single shot multibox detector to discriminate apples and a stereo camera to determine their three-dimensional positions. The arm harvests the apples by twisting the hand axis. The experimental results showed that this system detects more than 90.0% of the fruits and that the robot could harvest a fruit in 16 s.

However, in recent years, harvester robots based on multiarm configurations have gained attention. The idea is to improve the poor efficiency achievable with autonomous one-arm robotic harvesters by mounting multiple manipulators on a robotic platform and assigning a specific workspace to each manipulator to harvest [10]. For instance, the studies presented in [11], [12] focused on improving harvesting efficiency by developing algorithms that achieve the best distribution of fruits among the arms. In [12], the authors presented a four-armed kiwi harvester robot designed to operate autonomously in pergola-like orchards. The vision system uses deep neural networks and stereo matching to detect and locate kiwifruit in real-world lighting conditions. The proposal included a dynamic fruit scheduling system to coordinate the arms throughout the harvesting process. The performance evaluation results showed that the system was capable of successfully harvesting 51.0% of the total kiwifruit within the orchard with an average cycle time of 5 s/fruit.

In [13], the authors proposed a dual-armed cooperative approach for a tomato harvesting robot using a binocular vision sensor. The tomato detection algorithm combined the AdaBoost classifier and color analysis. The three-dimensional scene reconstruction was obtained in a simulation environment by using the point clouds acquired from a stereo camera. The achieved harvest success rate was 87.5%; meanwhile, the harvesting cycle time excluding cruise time, was less than 30 s.

A robotic harvesting system that performed recognition, approach, and picking tasks for aubergines was presented in [14]. The proposed machine vision algorithm combined a color segmentation process and a vertical dividing operation. To actuate the manipulator, they designed a visual feedback fuzzy control model that enables the manipulator end-effector

to approach the fruit from a distance of 0.3 m. The system achieved a successful harvesting rate of 62.5% and an aubergine-harvesting execution time of 64 s.

The aforementioned studies used more than one arm working independently; however, coordinating their behavior was not among the considered objectives. To fill this gap in robotic harvesting, this study proposes and validates planning and control algorithms for a dual-arm aubergine harvesting robot whose end-effectors operate cooperatively allowing it to reproduce complex human movements during harvesting tasks, e.g., with one arm pushing leave sideways while the second arm picks the fruit.

Vegetables such as aubergines must be harvested carefully to avoid damage, which is important for maintaining fresh-market quality and increasing product desirability. In recent years, the production trend for aubergines has undergone a significant increase within the European Union; Spain is the current leader in aubergine exports, although countries such as China and India are also notable in aubergine cultivation [15]. This increase shows the importance of aubergines agriculturally and economically. However, research studies that address the development of robotic harvesting systems targeting aubergines are scarce [14], [16]; the successful harvesting performance rates are low, and the harvesting time per fruit is high. These conditions motivated our interest in selecting this crop and the future possibilities that aubergine harvest automation can offer.

The remainder of this article is organized as follows. Section 2 describes the materials and methods used for the design and implementation of the proposed robotic harvesting system. Section 3 presents the image segmentation algorithm for detecting and localizing aubergines. Section 4 explains the planning algorithm that calculates the sequence of movements required to grasp and detach the aubergines. Section 5 discusses the design and implementation of the proposed dual-arm manipulation strategy when a fruit is occluded. Section 6 presents the results obtained from the experimental tests, and finally, Section 7 summarizes the main conclusions.

II. MATERIALS AND METHODS
This section describes the implemented dual-arm robotic platform and the proposed algorithms that reproduce complex human movements during harvesting tasks.

A. DUAL-ARM HARVESTER ROBOT
The hardware of the proposed harvester robot consists of a dual-arm robotic system and a sensor rig. The selected robotic arms are two Kinova MICO$^{\text{TM}}$ endowed with the Kinova Gripper KG-3 [17]. These arms are lightweight and feature low power consumption. Each robotic arm is composed of six interlinked segments providing 6 DoF with a maximum payload of 2.1 kg in mid-range continuous operation, which is an adequate load capacity for the gripper and for harvesting aubergines [18]. The grippers are underactuated with a set of three flexible fingers. The opening and closing movements

of the fingers are driven by three linear actuators, one for each finger, allowing objects to be grasped with a force of 40 N. The upper parts of the grippers can be equipped with a custom-made tool for cutting aubergine peduncles.

To exploit the capabilities of the dual-arm platform during precision harvesting tasks, the torso of the robot follows an anthropomorphic design [19]. Moreover, to achieve good robotic arm performances during dual manipulation, they are configured with right and left-handed configurations (see Fig. 1).
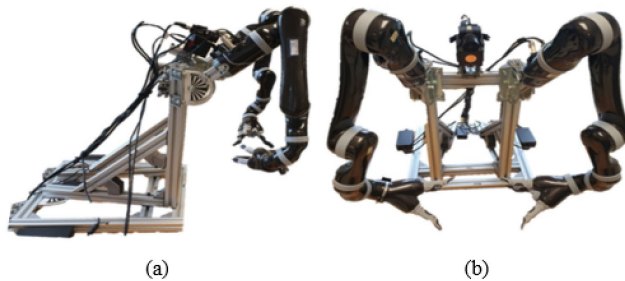

FIGURE 1. Prototype of the dual-arm harvester robot. a) lateral view b) front view.

The vision system consists of two cameras, a Prosilica GC2450C, which provides a high-resolution color image, and a Mesa SwissRanger SR4000, which provides a point cloud of the scene. The Prosilica GC2450C has a 5.0 megapixel resolution, is GigE Vision compliant [20], and incorporates a high quality sensor that provides superior image quality, excellent sensitivity, low noise, and a full-resolution frame rate of 32 fps. The Mesa SwissRanger SR4000 camera is a measurement device that captures 3D data of infrared (IR) light-reflective objects in the surrounding scene [21]. The distance measurement capability is based on the time-of-flight (TOF) principle. In nominal operation mode, an absolute accuracy of less than 0.01 m is achievable within a work range of 10 m at an acquisition rate of 50 frames per second.

Both cameras use a software triggering mode, which means that they wait for an "acquire" command before starting synchronized image capture. Both cameras communicate via Ethernet.

The software architecture system is implemented in the robot operating system (ROS) and formed by four modules, which are responsible for (i) image acquisitions from both cameras, (ii) detecting and localizing the aubergines in the robot's coordinate space, (iii) motion planning and (iv) control of the dual-arm robot [22]. At the heart of the architecture is the ROS master running on localhost, which makes it possible for nodes to find each other and exchange data. Each node has its own topics that can be used to publish or subscribe to messages. A node publishes data in a common space under a topic. Other nodes can use these data simultaneously by subscribing to that topic. As shown in Fig. 2, the system has six programmed nodes:

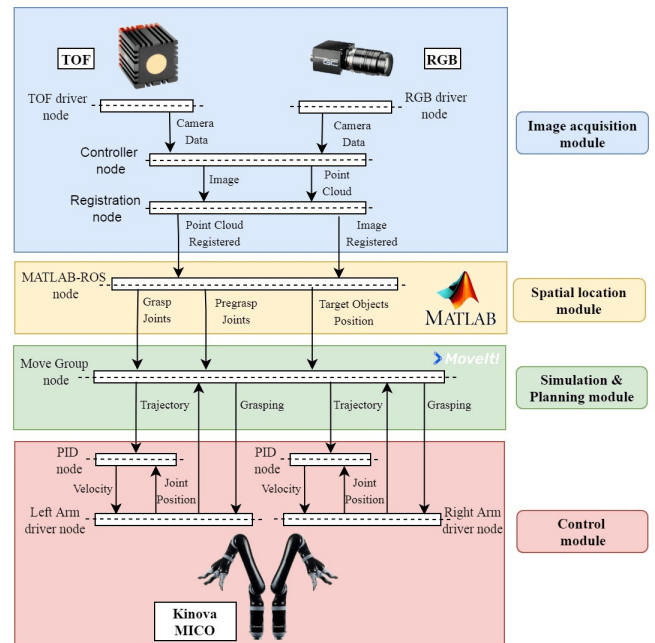- Two nodes within the image acquisition module for running both cameras (TOF and RGB) synchronously

FIGURE 2. Overview of the proposed software architecture.

and registering the color and range data in the same reference frame.
- The MATLAB-ROS node of the spatial localization module for recognizing the target objects, estimating their centroid positions, and calculating the inverse kinematics of the robotic arms.
- The Move Group node of the simulation and planning module, which is responsible for computing the necessary control inputs and sending the corresponding commands to the control module.
- Finally, the two PID nodes of the control module for running the joint of each arm according to the commanded control inputs.

B. METHOD

Fig. 3 summarizes the various steps of the designed and implemented decision-making strategy for automatic aubergine harvesting. Before starting, a reference model of the aubergine variety to be harvested is defined that includes the minimum size (minimum number of pixels in the image plane) that the fruit must occupy to fulfill the desired quality standards. Then, all the systems are initialized. The sensor rig proceeds with the synchronized acquisition of data from the effective field of view. The acquired color image and the point cloud data are registered due to the different pixel resolutions and the different camera fields of view. In this case, to reduce the computational load, the color data are mapped into the coordinate frame of the range data. Next, the registered color image and the point cloud are used as input to an image segmentation algorithm that detects aubergines based on four aspects: (i) reflectance measurements in the scene, (ii) the 3D positions of the candidate pixels in space, (iii) the sizes
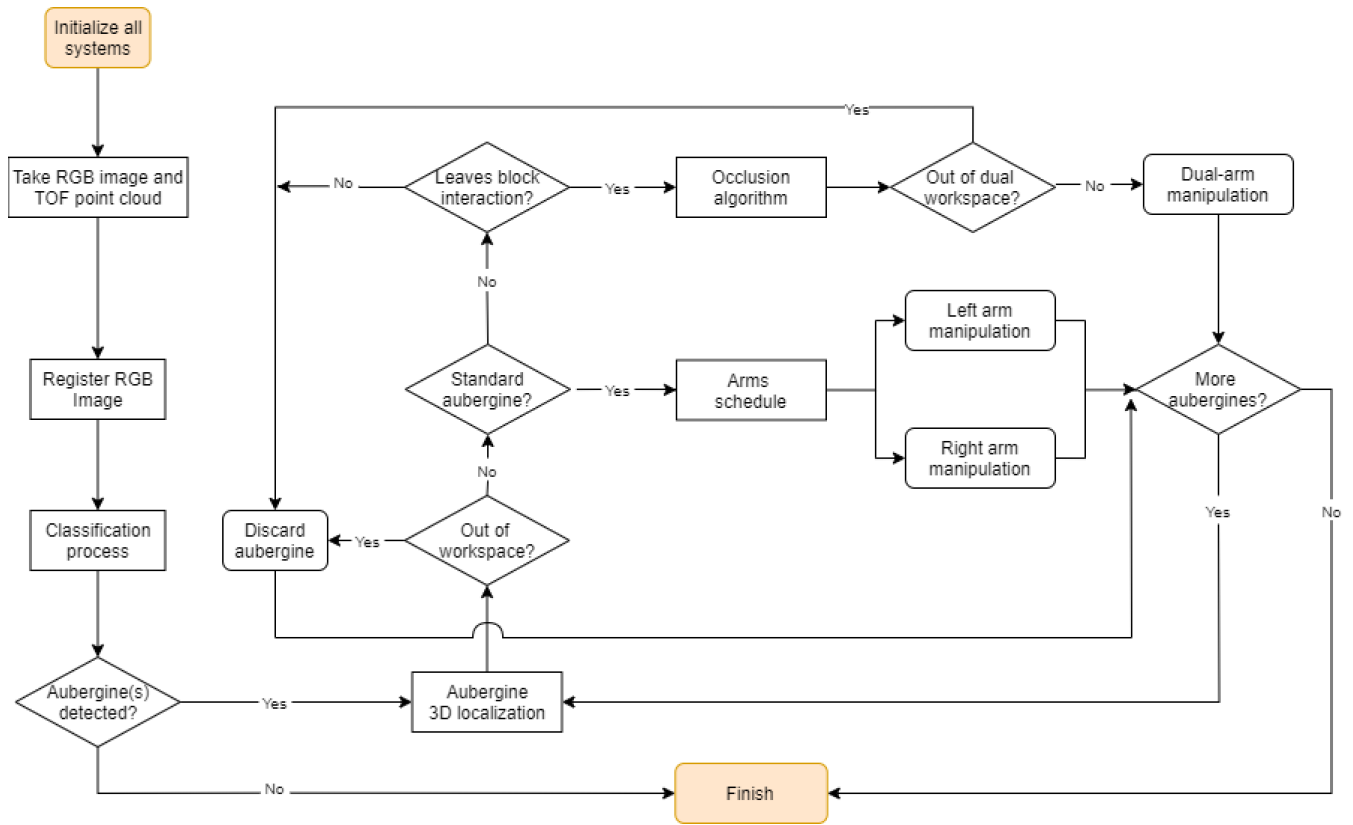
FIGURE 3. Overview of the main steps involved in the proposed aubergine harvesting cycle.

of the regions of interest, and (iv) interactions with blocking leaves. The point clouds of detected fruits that have a high visibility percentage and that meet the standards required for harvest are then used as input by a planning algorithm which, based on the workspace, determines the locations of the fruits, the arm configurations, and the movements necessary to grasp and detach aubergines. These movements may involve the simultaneous harvesting of two pieces of fruits, or harvesting with a single arm. In contrast, the point clouds of the fruits that have a low visibility percentage are further processed by the proposed occlusion algorithm, which plans collaborative arm behaviors to solve occlusion problems and implement dual-arm harvesting. The main algorithms involved in the proposed decision-making strategy are described below in more detail.

III. IMAGE SEGMENTATION
Image segmentation is a computer vision process that partitions a digital image into multiple regions to facilitate its analysis. Image segmentation is typically used to locate objects and boundaries in images. This process is trivial for humans; nevertheless, achieving robust image segmentation is still a challenge in computer vision applications because noise, low contrast, poor illumination and object boundary irregularities can lead to inaccurate results [23], [24].

The techniques commonly used in image segmentation are thresholding-based, gradient-based, region-based, edge-based, and classification-based [25]. Within the classification-based techniques, machine learning and deep learning algorithms play a relevant role by establishing relationships among multiple features to improve system efficiency. Each instance in every dataset used by the learning algorithms is represented by the same set of features. If instances are provided with known labels that represent the corresponding correct outputs, the learning process is called supervised. In contrast, in unsupervised learning, the training instances are unlabeled [26].

A number of surveys and reviews gathering the main advances on semantic segmentation have been presented in recent years. For instance, [27] provides an overview of broad segmentation topics including unsupervised and fully supervised methods as well as existing influential dataset and evaluation metrics. In [28] the strengths, weaknesses and major challenges of top image segmentation approaches are described. Deep learning for semantic segmentation is comprehensively reviewed in [29]–[31]. In [32] three categories of methods are reviewed and compared, including those based on hand-engineered features, learned features and weakly supervised learning. Last, weakly supervised image semantic segmentation is also reviewed in [33], [34].

In this study, the inputs and the desired outputs of the classification model are known; consequently, the selected learning method is supervised. The first step in supervised learning is to collect the dataset and determine which features

are the most informative. In this study, the dataset consists of 1,753 aubergine samples acquired under different lighting conditions, and the feature used is the colors of the different scene elements. Color is a popular visual cue in machine vision tasks, and it is an appropriate choice for a discriminative feature because vegetables tend to have different reflectance properties than do the foliage and branches around them.

However, instead of using the original R, G, and B values directly, we introduce color transformations before applying the segmentation algorithm to reduce its sensitivity to changing illumination conditions. These transformations quantify the intensity differences between the red and green channels (R-G) in the RGB color model and the hues in the HSV (hue saturation value) color model [35]. These images are then used as inputs for the segmentation process. The proposed image segmentation algorithm consists of three parts: a support vector machine (SVM) (which is a pixel-based classifier), a watershed transformation and the corresponding point cloud extraction.

To design the pixel-based classifier, and considering the agricultural scenario of interest, we designed four classes: aubergines, leaves, branches and the scene background. We tested different algorithms to find the model that best fits the data. A dataset was randomly selected for training these algorithms. Table 1 lists the obtained results. Clearly, the algorithm that best suits the data is the SVM cubic algorithm, which achieved a success rate of 97.4%. Consequently, this algorithm was selected for the segmentation process.

TABLE 1. Accuracy scores of some of the tested models.

Model Type	Accuracy
Cubic SVM	97.4%
Quadratic SVM	97.3%
Fine KNN	96.6%
Linear SVM	94.4%
Fine Tree	94.4%
Boosted Trees	94.4%

SVM is a supervised machine learning algorithm widely used in classification and pattern recognition tasks. An SVM chooses the decision boundary that minimizes the generalization error by selecting the hyperplane that provides the maximum separation or margin between the classes [36].

SVMs are well suited to learning tasks where the number of features is large with respect to the number of training instances, and they tend to perform much better when dealing with multiple dimensions and continuous features. Therefore, a large sample size is required to train an SVM to achieve its maximum prediction accuracy. SVMs also perform well when multicollinearity is presented and when a nonlinear relationship exists between the input and output features [26].

After the pixel classification (see Fig. 4), the aubergines can be discriminated from the remaining elements in the scene (leaves, branches and background). Next, to separate adjacent aubergines that appear as a single blob, we apply a

procedure based on the watershed transform and the minima imposition technique [37], [38]. The watershed transformation is an effective morphological tool that treats an image as a topographic surface, providing catchment basins and watershed ridge lines by assuming that objects are characterized by a homogeneous texture (and hence a weak gradient). First, noise should be removed to eliminate small dots that should be in the aubergine class. Then, the watershed transform of the image is computed. The watershed transform is known for its tendency to oversegment an image because each local minimum becomes a catchment basin. One solution to avoid this problem is to filter out tiny local minima and then modify the distance transform: this process is called minima imposition [39]. After these steps, the watershed transform is computed again, and the resulting watershed ridge lines are utilized to separate the adjacent aubergine blobs. Based on these blobs, the point clouds of the detected aubergines are extracted, and their corresponding centroids are estimated.

However, because this process is performed with the camera provided data, a transformation must be performed from the camera coordinate system to the robot base, as follows:

$$pixel_{coord} \cdot T_{pixel}^{cam} \cdot T_{cam}^{end} \cdot T_{end}^{base} = p_{base} \qquad (1)$$

As shown in (1), first, the pixel coordinates (x, z) are transformed into the camera coordinates, using the camera projection matrix. After obtaining the planar projective coordinates, the y axis distance provided by the TOF camera is added. Using the camera-robot calibration proposed by Taylor [40] the transformation matrix between the camera and the end-effector robot is extracted. Finally, by applying the transformation of the end-effector robot to its base, the 3D localization of each aubergine with respect to the end-effector is procured.

At this point, the planning algorithm discards any aubergine that is outside the workspace of both arms. Then, a new decision criterion based on the average fruit size is introduced into the process. It is well known that vegetables must fulfill various requirements to reach the quality level required by fresh markets. One such requirement is average size. All aubergines should be approximately the same size when harvested.

Because the cameras are fixed on the robot's torso, the visible area of the aubergines is estimated by counting the number of pixels in each separate blob. Nevertheless, the area of a region in an image changes according to the distance from the camera to the object. Consequently, we apply a correction distance factor to all the aubergine blobs. This correction is applied according to the field of view of the camera, which is the part of the world visible to the camera at a particular spatial position and orientation. This view is most often expressed as the angular size of the view cone, that is, as a view angle. From its technical specifications, the TOF camera has a field of view of 69° × 56° (see Fig.5-(a)).

Using the cone of the field of view and trigonometric rules, the real area of each aubergine can be calculated independently of its distance to the camera. To calculate the true

FIGURE 4. Segmentation process: (a) captured image; (b) classification of all the scene elements; (c) aubergine classification; (d) leaf classification.

area, the length values of the major and minor axes of the aubergine regions are also necessary. Then, the tangent of α (see Fig. 5-(b)) is given by:

$$\tan(\alpha/2) = \frac{H/2}{d_z} \qquad (2)$$

where α is the angle corresponding to the vertical field of view (56°); d_z is the z-component of the distance from the extracted object to the optical center of the TOF camera; and H is the height of the complete image at that distance. The latter term is an unknown value.

After calculating the height of the entire image (in meters), the proportional factor between that height in meters and the total height of the image in pixels (144 px) can be obtained. Next, some manipulations are conducted. For simplification purposes, the shapes of the aubergines are approximated to an ellipse; this approach is beneficial because it requires less processing time and well matches the overall aubergine shape. In addition, it can be generalized that aubergines hang upright due to their weight. Therefore, the major axis of the ellipse corresponds to the length of the aubergine. Consequently, by applying a proportional factor, the length of the aubergine in meters can be calculated.

In addition, the width of the aubergine is needed to calculate the area of the ellipse. Following the same process described above for the height, but with the trigonometry obtained from Fig. 5-(c), the width is obtained as follows:

$$\tan(\beta/2) = \frac{W/2}{d_z} \qquad (3)$$

After calculating the height and width, the real area of the aubergine can be extracted using the equation of the area of an ellipse; this result is independent of the distance.

As mentioned above, aubergines are harvested when they have reached an optimal size. Because of this, the optimal size is used to discriminate aubergines with the quality required by fresh market, from those that do not fulfill the requirements. Thus, the estimated size of the detected aubergines is compared with a predefined template to calculate the percentage of visibility of each fruit. Depending on the percentage of visibility, the aubergines are categorized as either whole or partially occluded (those whose visibility percentage is
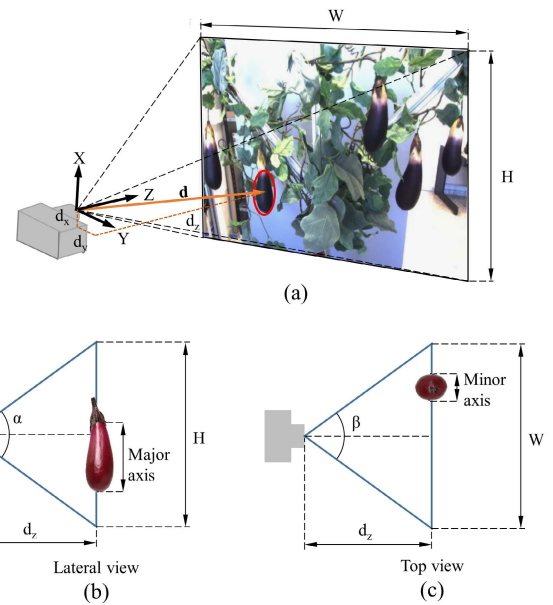

FIGURE 5. Camera views: (a) camera field of view; (b) obtaining the aubergine major axis; and (c) obtaining the aubergine minor axis.

below 80% of the template model), which imply different manipulation strategies.

IV. PLANNING ALGORITHM

This section is devoted to the manipulation of aubergines that lie within the dual-arm workspace and fulfill the visibility criteria. Fig. 6 shows a visualization of the workspace of both arms in 3D. The idea is that each robotic arm should be assigned to specific aubergines to harvest them in a collaborative manner. The decision-making process is based on the 3D position of the aubergine centroid, which is its center of mass.

To fully exploit the capabilities of the dual-arm configuration, the robot can grasp two aubergines using both arms simultaneously. However, one of the objectives for this robotic system is to achieve effective cooperation between the arms to increase picking efficiency while avoiding arm collisions. To solve this problem, a harvesting schedule is calculated that minimizes the collision opportunities for the
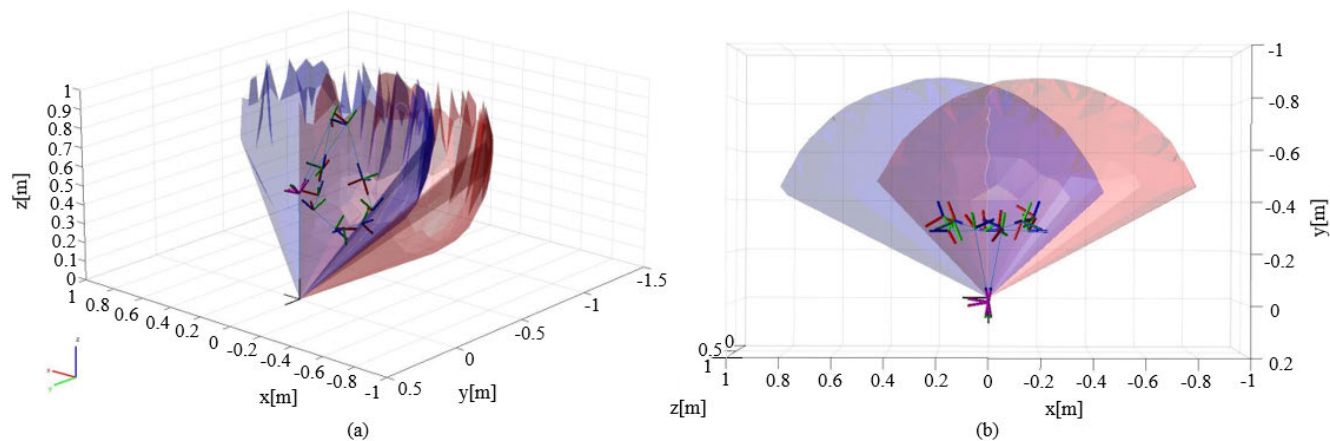
FIGURE 6. Workspace of the robotics arms (blue and red colors represent the workspaces for the left and right arms, respectively): (a) workspace in 3D and; (b) workspace in the *x-y*-axes.

robotic arms. In the harvester, as illustrated in Fig. 6-(b), based on the x axis, the picking area in the camera views is divided equally into a right arm section (section 1, in red) and a left arm section (section 2, in blue). Because the gripper is designed to pick aubergines in parallel with the $x - y$ plane (see Fig. 6-(b)), one requirement is that the robot should pick aubergines from the right to the left in section 1. In addition, because the robot's torso is designed following an anthropomorphic configuration, the arms should grasp the aubergines in a human-like fashion. Therefore, the right gripper is given a favorite orientation parallel to the y axis; if this orientation cannot be reached for the aubergine position, the orientation is changed according to variations of $\pi/4$ until it achieves an orientation parallel to the x axis, where the right gripper opening faces the positive direction. With every change in orientation, the final position of the arm is recalculated until a feasible position is found. The process is the same for the left arm; however, in this case, the left arm will start picking aubergines from the left part of section 2 to the right part, and the final calculated orientation will be parallel to the x axis but with the left gripper facing the negative direction.

Following this procedure prevents the grippers and arms from touching or from moving the central aubergines. During the process of inputting the detected aubergines into the algorithm, the first goal is to determine the picking sequence for the arms to maximize the simultaneous picking period and avoid possible collisions. The default picking sequence for both arms flows from the extremes to the middle of the workspace. However, when the number of aubergines in the section 1 is equal to or less than in section 2, it is better for both arms to pick aubergines simultaneously. With the exception of the aubergines that are in the dual-arm workspace, in this section, it is necessary to check the distances among the selected aubergines to avoid collision between the two arms. To maintain a safe distance, the aubergines must be at least 0.16 m apart. When aubergines are closer than this safety distance, they will be picked only with one arm - the

arm for which more aubergines are available. If the number of aubergines that can be picked simultaneously is equal for both arms, the central aubergines can be harvested with either arm. Algorithm 1 summarizes the different steps described above.

After an aubergine is collected, the arm must move to a release position; therefore, the initial positions of the arms are always the same.

Once the picking sequence is set, it is necessary to perform the planning of the trajectories to avoid collisions. For this purpose, a virtual scene is created, which is used to represent the world around the robot, as well as the state of the robot itself. In the scene, obstacles such as the structure of the robot and the floor, are included and considered by the motion planner to avoid collisions of the robotic arms with elements in the real world. To keep the virtual scene as similar as possible to the real world, aubergines are also introduced by using the point cloud of their corresponding regions of interest. Fig. 7 shows an example of a manipulation scene captured during the experimental tests.

This planning scene is developed in MoveIt!, an open source robotics manipulation platform [41], which works with motion planners from multiple libraries through a plugin interface. In this case, the motion planner is configured using the Stochastic Trajectory Optimization for Motion Planning (STOMP), which is a probabilistic optimization framework [42]. STOMP produces smooth well-behaved collision free paths within reasonable times. The motion planning generates noisy trajectories, which are then combined to produce an updated trajectory with a lower cost. This cost function combines the cost of the obstacles and the smoothness and it is optimized in each iteration. The trajectory is then generated in response to the motion plan request using the robot's current state and the target, but also checking collisions with the obstacles, including self-collisions.

The resulting trajectory is formed by several waypoints and each of them contains the position, velocity, and acceleration for all of the joints of both arms, as well as the start time

Algorithm 1 Algorithm to Address the Motion Planning for the Arms

---

Input: List of the detected aubergine(s) *det*
Output: Lists of picking *dualArm*, *leftArm*, *rightArm*

1: **for** each aubergine *i* in *det* **do**
2:     Extract the centroid C_i
3:     **if** $C_i(x) > 0$ **then**
4:         Aubergine added to *leftArm*
5:     **else**
6:         Aubergine added to *rightArm*
7:     **end if**
8: **end for**
9: Sort *leftArm* acording to $C(x)$ from major to minor
10: Sort *rightArm* acording to $C(x)$ from minor to major
11: **for** $i = 1$ to $min(size(leftArm), size(rightArm))$ **do**
12:     Pair ($leftArm_i$, $rightArm_i$) added to *dualArms*
13:     Remove $leftArm_i$ from *leftArm*
14:     Remove $rightArm_i$ from *rightArm*
15: **end for**
16: **for** each pair of aubergines ∩ dual-arm workspace in *dualArm* **do**
17:     **if** distance between them < safe distance **then**
18:         **if** *leftArm* is empty **then**
19:             Both aubergines added to *rightArm*
20:         **else**
21:             Both aubergines added to *leftArm*
22:         **end if**
23:         Remove pair of aubergines from *dualArm*
24:     **end if**
25: **end for**
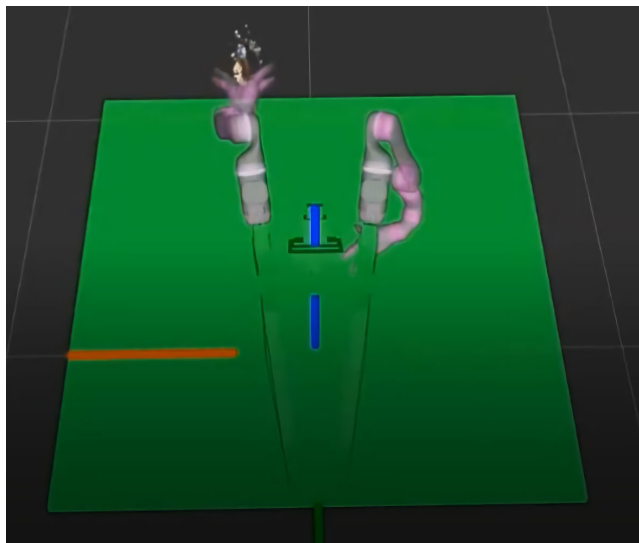26: **return** *dualArm*, *rightArm*, *leftArm*

FIGURE 7. A virtual manipulation scene captured during the experimental tests.

of the next trajectory waypoint. Finally, the waypoint positions of the trajectory are used for the proportional–integral–derivative (PID) controllers to provide the motion execution command to the robot

V. OCCLUSION ALGORITHM

If an aubergine is not marked as a candidate to be harvested, the next step is to check whether it is occluded by leaves or is too small to fulfil the area criteria.

To determinate whether an occlusion exists, the algorithm checks whether leaves are located in a space that a ready-to-harvest aubergine would occupy by constructing an overlapped area. To accomplish this task, the aubergine template model is placed by matching its centroid with that of the visible area extracted in the previous step. In addition, the orientation is also considered for overlapping in the template model.

After overlaying the aubergine template model, a new image processing procedure begins. The next step is to calculate the intersection between the template model and the leaves within its area. This process is performed from the 2D image. The goal is to achieve a high compute speed. Although cases exist in which overlapped leaves may be located far from the aubergines, the algorithm still finds an intersection with them. The solution for these cases is to include the distance from the centroid of these intersections to the camera. In this way, the algorithm can discriminate among leaves that could be an occlusion source and those that are far from the aubergine.

At this point, the target aubergine will be ignored by the harvesting process if no leaf-aubergine intersection is detected based on the idea that this aubergine does not meet the criterion of area because it is too small to be harvested.

Some aubergines may have several sources of occlusion. In this situation, the criterion used is that the larger intersection causes the biggest occlusion problem; therefore, it is the occlusion addressed by the system.

The next problem that arises is to schedule the arms. The robot's bimanual capabilities can be employed to manage leaves with one arm while the other grasps the aubergine. Therefore, in this process, both arms are used in the same workspace; the system functions only for occlusions localized to the workspace area shared by both arms.

In addition, to avoid occlusions that may be generated during movements of the arm assigned to move the leaves aside, we have considered several possible conditions. First, the direction of the vector that joins the centroid of the visible part of the aubergine and the centroid of the intersection must be calculated. The direction of this vector in the *x* axis will determine the arm used to move the leaves. In this way, the system ensures that the aubergine will not be occluded by the arm. The final step is to calculate the distance that the arm should move the leaves to obtain a clear view of the entire aubergine. Thus, the algorithm calculates another point along the line that joins these two points. This point must be separated from the intersection centroid by at least 0.15 m to ensure that the entire aubergine is visible and avoid gripper occlusions.

The different steps of the proposed algorithm are visualized in Fig. 8 and summarized in Algorithm 2. In Fig. 8, the cyan irregular line shows the contour of the detected aubergine
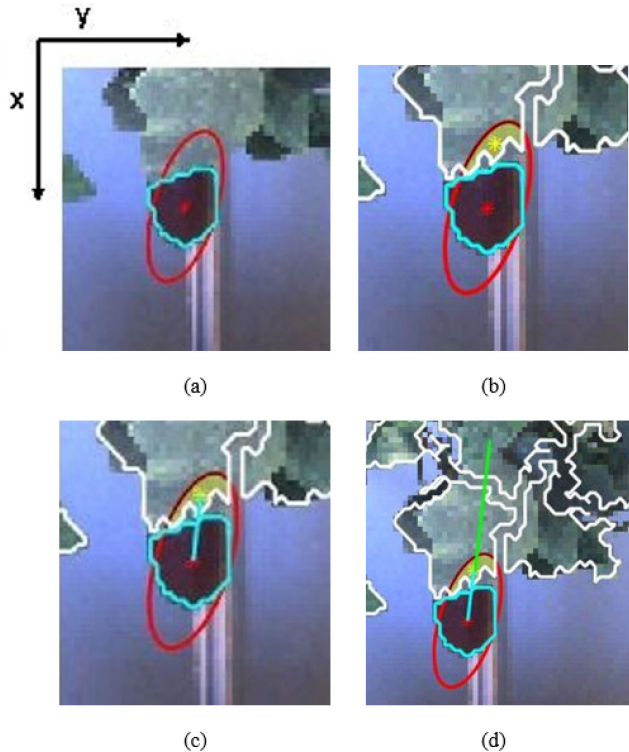
FIGURE 8. Image processing performed for aubergines that do not fulfill the area criteria; (a) Overlay of the standard aubergine (red) using the centroid and orientation of the partially occluded aubergine (cyan); (b) calculating the intersection area between the standard aubergine and the leaves block (yellow); (c) calculating the vector that joins the centroid of the aubergine with the intersection (cyan); (d) calculating the final position where the arm pushes the leaves to move them apart (green).

blob; the red asterisk represents the centroid estimated from the detected aubergine blob; the white irregular lines correspond to the contours of the detected leave blobs; the red ellipse corresponds to the model template overlapping over the occluded blob; and the green line represents the direction vector along which the robotic arm moves to sweep the leaves aside and remove the occlusion.

Another consideration is the orientation of the arm that will move the leaves. In an experimental phase, we determined that the best way to move the leaves is to proceed with the arm parallel to the y axis and with the gripper closed. Consequently, the arm simply pushes the leaves away. In this way, the system avoids having to grasp the leaves with the gripper, which reduces the complexity of the movement.

After displacing the leaves, a new centroid is calculated for the entire aubergine so that the peduncle can be cut correctly to prevent damage to the vegetable.

Finally, it is important to note that the proposed strategy is the same in all cases with occlusions because the point of contact with the aubergine is the same, regardless of the distance with the block of leaves. Therefore, since only a displacement of the leaves is performed and the aubergines are not manipulated in the process, they are not damaged. In addition, the fingertips of robot grippers have a deformable

---

Algorithm 2 Algorithm to Address Possible Occlusions

Input: List of the aubergine(s) that not fulfil the area criteria oc, list of all the leaves block l
Output: The displacement vector dir

1: **for** each aubergine i in oc **do**
2:     Extract the centroid C_i
3:     Extract the orientation O_i
4:     Standard aubergine overlapping S_i, with the same C_i and O_i
5:     **for** each leaves block j in l **do**
6:         **if** $\exists\, S_i \cap l_j$ and $l_j(z) < C_i(z)$ **then**
7:             Occlusion of oc_i with l_j
8:             Calculate the area of $S_i \cap l_j$, A_{ij}
9:         **else**
10:             Classify oc_i as discarded aubergine
11:             **break**
12:         **end if**
13:     **end for**
14:     Extract the centroid C_{ij} of $max(A_{ij})$
15:     Calculate $dir = C_i(x) - C_{ij}(x)$
16: **end for**
17: **return** dir

---

rubber that prevents possible damage to the aubergines during contact events.

VI. EXPERIMENTAL RESULTS

A. EXPERIMENTAL SETUP

The experiments were conducted under laboratory conditions at the Centre for Automation and Robotics using the dual-arm robotic platform and the software architecture described in Section II. Aubergines (*Solanum melongena*) of the variety named "Thelma" distributed over a plant model were selected for the experimental tests. These aubergines were sourced from a greenhouse in Almería, Spain."

To validate the different algorithms that comprise the proposed decision-making strategy, we conducted 90 experiments to demonstrate the performance of the robotic harvester in the most common real-world situations. The experimental results provide valuable information on the advantages of the system and on the challenges we face in improving the robotic harvester. To perform an exhaustive analysis of the extracted data, the results are separated into the achievements of the sensor rig, the bimanual capacities of the robot provided by the planning algorithm and the results of the novel occlusion algorithm presented in this article. Finally, we present achievements of the complete system.

B. EVALUATION OF THE IMAGE SEGMENTATION ALGORITHM

To evaluate the output of the image segmentation algorithm, the ground-truth data were carefully produced by manually labeling the pixels that belonged to the visible areas of the aubergines. Then, the aubergines detected by the algorithm

were compared with the ground truth data, and the detection performance was evaluated at the pixel level in terms of the true-positive rate (TPR), false-positive rate (FPR) and false-negative rate (FNR) [35]. The mean values obtained from all the analyzed scenes as well as the minimum and maximum values are presented in Table 2.

TABLE 2. Performance assessment at the pixel level of the proposed detection algorithm.

Values	TPR	FPR	FNR
Mean	85.32%	0.05%	14.68%
Minimum	77.90%	0.01%	9.28%
Maximum	90.71%	0.18%	22.10%

The performance evaluation results at the pixel level show that the proposed detection algorithm exhibits a high hit rate of 85.32%, a low FPR of 0.05% and an acceptable FNR of 14.68%. The poor FNR values generally occur at the edges of the aubergines; the system identifies these pixels as a different class due to the high contrast between the color of the aubergines and the background.

In addition, the proposed detection algorithm is evaluated at the aubergine level in terms of recall, precision and F-score (the weighted harmonic mean of the test's precision and recall). In this case, instead of counting pixels, the aubergines are counted as units. The results of this analysis can be seen in Table 3.

TABLE 3. Performance evaluation at the aubergine level.

Recall	Precision	F-score
88.10%	88.35%	0.878

At the fruit level, the TPR of aubergines detected (Recall) by the proposed algorithm is 88.10%, which indicates that the algorithm fails to detect only a small number of targets. From the results, most of the errors in undetected aubergines are caused by the watershed transformation, which in some cases does not separate the blobs into a correct number of available aubergines due to lighting conditions. In addition, the average precision provided by the detection algorithm was 88.35%, which indicates a slightly higher number of false positives compared to the number of false negatives. Such misclassifications typically occur due to shadows produced by the leaves.

Furthermore, the proposed algorithm achieves an F-score of 0.878, which is a competitive value compared with methods used for harvesting other fruits. For example, [43] presented a system for detecting mangoes and obtained an F-score of 0.881, while [44] achieved an F-score of 0.838 when detecting sweet peppers and rock melons. Considering these scores, the F-score obtained by the proposed algorithm for aubergine detection has a competitive advantage over the other promising approaches; the competitors require more computation but do not differ substantially in terms of accuracy.

In the following, we present two tests that illustrate the operation of the image segmentation algorithm.

The first test represents a simple case in which the scene is composed of isolated aubergines without no occlusions. Fig. 9 shows (a) the registered RGB image, (b) the pixel-based classification map provided by the algorithm, (c) the ground truth image and (d) the detected aubergines. The output of the segmentation algorithm is quite similar to the ground truth image; the correct classifications of the four aubergines are visible in the image along with some inaccuracies in the pixels at the edges of the aubergines.

In the second test (see Fig. 10), we tested the ability of the system to address a common situation in image segmentation: two overlapping targets. The overlapping aubergines may be at the same distance or one may be in front of the other, causing them to appear connected in the image. As explained above, to address this type of situation, the system incorporates the watershed transformation to separate the blobs of different aubergines. Fig. 10-(d) shows a correct performance of the proposed algorithm, which is capable of separating the detected blobs and thus discriminating between two different aubergines.

C. EVALUATION OF THE PLANNING ALGORITHM

To assess the performance of single and dual-arm harvesting, three cases are studied in this subsection. First, harvesting with a single arm, which grasps the only aubergine available in the scene. Second, harvesting by capitalizing on the movement capabilities of both arms to pick two aubergines using both arms simultaneously. The final case involves harvesting two aubergines with the same arm. Table 4 lists the collected times for these three cases. These results were obtained by executing the image-processing algorithm and the inverse kinematic calculations in MATLAB and using MoveIt! to plan the trajectories and the execution of the real movements of the robot. The computer used was equipped with an Intel i7-4790 processor running at a clock speed of 3.6 GHz and 8 GB of RAM. The times shown were the averages of executing ten trials for each case.

The image processing time includes the time spent to register the RGB image, obtain the pixel-based classification map, segment the aubergines that appear in the scene, and obtain their locations in 3D space, as well as the time dedicated to arm allocation in the dual-arm manipulation case. The inverse kinematic was calculated using the Robotics System Toolbox in MATLAB. Finally, the action time includes the time required for robot movements; this also includes the time for calculating the trajectories. The motion sequence involves four actions: movement to the pregrasp position, grasping, postgrasping and release to place the aubergines in the collection box.

The results in Table 4 show that the time dedicated to image processing is similar in all three cases. Nevertheless, in cases where there are two aubergines, the time is longer because the planning algorithm must assign the correct arm for each aubergine.
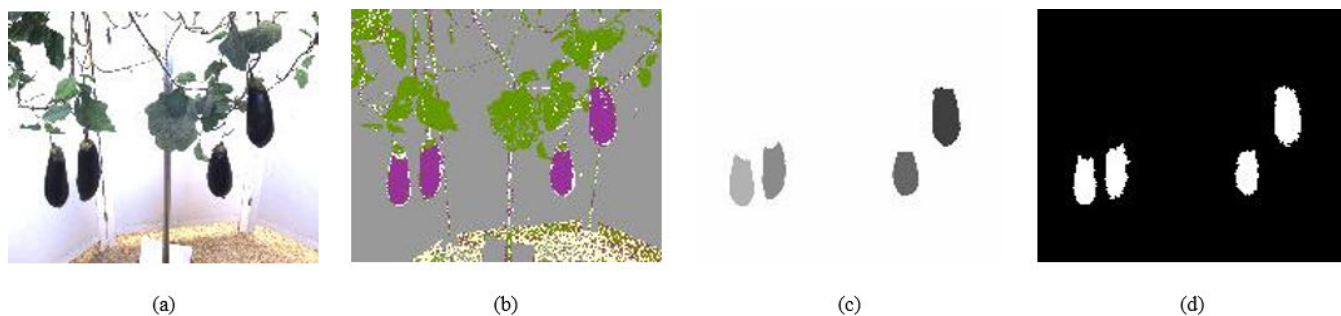
FIGURE 9. Experimental result - isolated aubergine scene: (a) registered RGB image (b) pixel-based classification map (c) pixel labeling (d) output of the aubergine class.

FIGURE 10. Experimental result - connected aubergine scene: (a) registered RGB image (b) pixel-based classification map (c) pixel labeling (d) output of the aubergine class.

TABLE 4. Comparison of one arm and dual-arm performance.

Procedures	One-arm One fruit	Dual-arm Two fruits	One-arm Two fruits
Image processing	0.91 s	1.08 s	0.93 s
InvK	5.76 s	12.33 s	14.57 s
Action	36.23 s	39.66 s	67.84 s
Total	42.90 s	53.07 s	83.34 s

Now, focusing on the time corresponding to the computation of the inverse kinematics, this is clearly highest in the two-aubergine cases, approximately double that of the single-aubergine cases. This outcome is logical because these cases require two different positions of the end-effector to be calculated.

Finally, considering the time spent on the movement of the harvester robot, which includes the trajectory planning using the STOMP method [42], it can be found that an increase occurs for the dual-arm manipulation compared to the single arm manipulation for one aubergine. This is because the system needs to calculate two paths to produce a cooperative movement and check for possible collisions between the two arms. These conditions increase the complexity of the trajectory estimations, resulting in a greater computational load. However, because the proposed algorithm avoids collision by dividing the workspace for each arm at the middle, the time spent checking for collisions is essentially negligible. This minimal time increase is noteworthy because the robot's productivity is doubled when using both arms for picking.

Moreover, in comparison to the time required by a single arm to pick two aubergines, the results are plainly more advantageous for dual-arm manipulation.

Overall, the dual-arm configuration represents a significant improvement to the system that increases productivity because it can collect a larger number of fruits in a shorter period compared using only a single arm.

D. EVALUATION OF THE OCCLUSION ALGORITHM

This section is considered the most important aspect of this study because it assesses the performance of the algorithm that enables the dual-arm robot to reproduce complex human movements during harvesting tasks.

For the occlusion algorithm to perform correctly, image segmentation is a significant step. The results obtained from the image segmentation algorithm for the aubergine and leaf classes are presented below.

The assessment is performed at the pixel level by comparing the images obtained from the segmentation algorithm with the ground truth data produced by the manually labeled pixels. To analyze this case, the labeling of the leaves for the ground truth data correspond to those responsible for generating the occlusion and the leaves adjacent to the occluded aubergine; an example of the segmentation is shown in Fig. 11. The metrics used to evaluate the detection algorithm performance include TPR, FPR and FNR. The mean, minimum and maximum values obtained from 16 analyzed scenes containing one occluded aubergine are presented
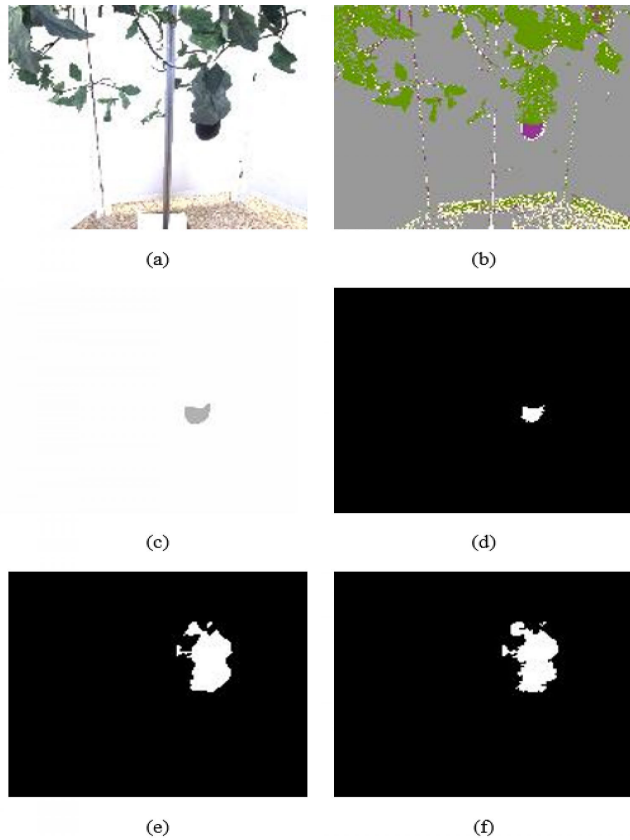
FIGURE 11. Experimental results - occluded aubergine scene: (a) registered RGB image (b) pixel-based classification map (c) pixel labeling of the aubergines (ground truth data) (d) output of the aubergine class (e) pixel labeling of the leaves (ground truth data) (f) output of the leaf class.

in Table 5 for the aubergine class and in Table 6 for the leaf class.

TABLE 5. Performance assessment at the pixel level of the proposed detection algorithm for occluded aubergines.

Values	TPR	FPR	FNR
Mean	75.51%	0.004%	24.49%
Minimum	52.94%	0.00%	12.97%
Maximum	87.03%	0.02%	47.06%

The average TPR obtained for the aubergines is quite satisfactory, considering that only occluded aubergines were considered in the estimations. Generally, these occluded aubergines are affected the most by shadows. The TPR for leaves is high because the leaves are more visible than are the aubergines in these cases. Therefore, the detection rates obtained are sufficient for the proposed algorithm to operate correctly.

From the execution times presented in Table 7, it can be observed that the time spent on the occlusion algorithm is small compared to the rest of the times. In contrast, the time dedicated to calculating the inverse kinematics is greater because the system must perform the calculations required for the movements that enable the robot to move the leaves aside

TABLE 6. Performance assessment at the pixel level of the proposed detection algorithm for leaves that create occlusion.

Values	TPR	FPR	FNR
Mean	91.21%	0.21%	9.79%
Minimum	78.06%	0.06%	0.52%
Maximum	99.48%	0.52%	21.94%

TABLE 7. Average times for scenes with occlusions.

Procedures	Time (s)
Image processing	0.88
Occlusion algorithm	0.11
InvK	12.16
Action	73.98
Total	87.13

and pick the exposed aubergines. However, to reduce the time dedicated to these calculations, we limited the ranges of the joint angles to find solutions most similar to those implemented by humans during harvesting. Therefore, notably, the computing time of the inverse kinematics is similar to that required for the dual-arm manipulation.

In this case, because the robot manipulates the leaves, we divided the success rates into the correct harvesting of an aubergine and the correct movement to move the leaves aside. The success rate for harvesting an aubergine is 93.75%, while the success rate for moving the leaves aside is 81.25%. After studying various scenes, most of the leaf-movement failures occur due to the inability of the gripper to contact them appropriately, causing the leaves to return to their original positions. Other errors stemmed from incorrect scheduling of the arms, producing a similar failure, in which the leaves slide off the gripper. This is because the image detection system does not consider the point where the leaves are attached to the aubergine plant. Consequently, the leaves are scheduled to be moved with one arm to eliminate the occlusion but that movement is not sufficient to keep the leaves from occluding the aubergine. Therefore, this problem can be addressed by improving the image segmentation algorithm.

E. COMPLETE SYSTEM EVALUATION

To evaluate the performance of the complete system, we executed 10 complete scenarios with the different cases presented above. Fig. 12 shows a scenario containing three isolated aubergines that must be grasped with one arm, and a partially occluded aubergine that uses the dual-arm manipulation capabilities. In addition, the scenario includes one aubergine that is not ready to be harvested due to its size. The two main metrics were used to test the harvester robot include the success rate and the picking cycle time; these represent the harvesting accuracy and speed, respectively. Failure cases were recorded and analyzed to identify challenges to be addressed in future versions of the system.

Table 8 shows the harvesting success rates for the three types of proposed manipulations. The average success rate is 91.67%. The table shows that cases with more unpicked
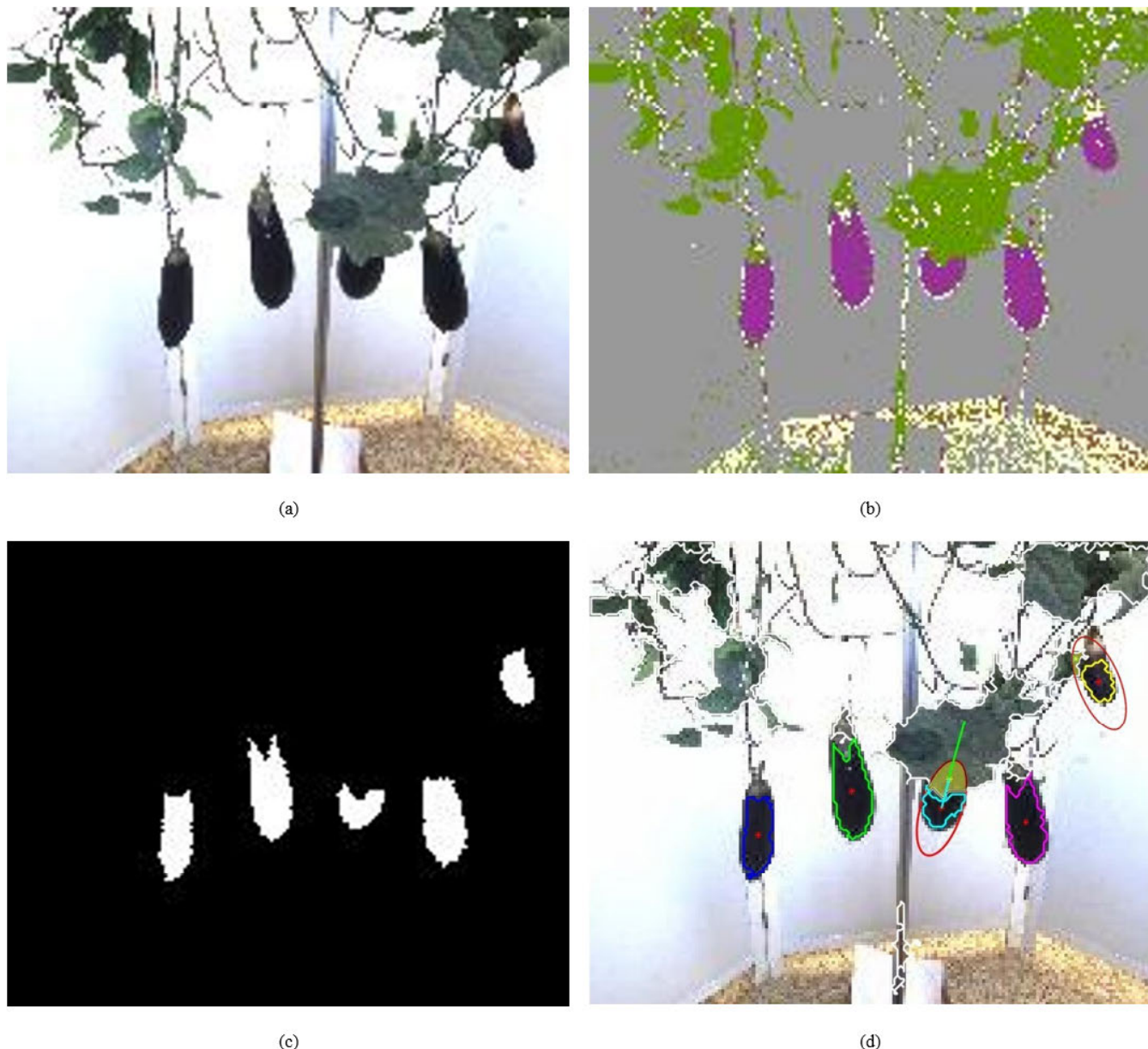
FIGURE 12. Experimental results - Test 4: a) registered RGB image; b) pixel-based classification map; c) ground truth data; d) output of the aubergine class. The red asterisks represent the centroids estimated from the detected aubergine blobs, the irregular colored lines show the contours of the detected aubergine blobs, the white irregular lines correspond to the contours of the detected leaf blobs, the red ellipse indicates the model template superimposed over the occluded blob, and the green line represents the direction vector that the robotic arm follows to brush the leaves aside and remove the occlusion.

TABLE 8. Success rate of the harvesting robot.

Condition	Success on the first attempt	Unpicked	Success rate (%)
One isolated piece (one arm)	10	0	100%
Two isolated pieces (dual-arm)	19	1	95%
One piece occluded	8	2	80%
Average			91.67%

aubergines correspond to the scenarios with occlusions. The failure cases are caused by the vision system, which does not recognize some aubergines due to the low visibility percentages they present. The success rate for isolated aubergines is very high; only one failure case occurred when the gripper grasped the aubergine only with the fingertips, and the fruit fell to the ground before the arm reached the release position. A special case in these scenes is the treatment of small aubergines. These aubergines were identified as occluded in two events because they were surrounded by leaves. This problem can be addressed by taking a second image after the leaves have been moved aside. To evaluate the cycle picking time, we carried out a review using time measurements that other agricultural research studies have included; we found different configurations were used

depending on the considered crop. For example, [45] focused on strawberry harvesting and divided the cycle into perception time and harvesting time but without including in the latter the manipulator configuration time required to drop the individual fruits. Others, dedicated to collecting tomatoes, included the complete working procedure, including the time required to place the fruit into the collection box [46], [47]. For our system, the cycle picking time includes the release time because due to the weight of the aubergines, the arm must deposit grasped aubergines into a collection box before starting a new grasping motion.

Most of our harvesting robot's time is spent in the manipulation process. The average time for the perception process is 0.81 s, including image registration, segmentation, 3D location, planning algorithm and dealing with occlusions. This time can vary depending on the number of targets included in the captured image as well as on the scene complexity. The harvesting time, including the time required for the arm to travel to the aubergine, the picking time and the release time, is 26.2 s on average. This time was obtained in a scene containing five aubergines with the characteristics previously discussed.

This average harvest time is considered satisfactory; to the best of the authors' knowledge, this study is the first time that a harvesting process has been proposed that uses two arms cooperatively in an unstructured environment similar to a human being.

VII. CONCLUSION

This article presented a dual-arm robotic system and proposed a decision-making strategy designed and implemented for automatic aubergine harvesting in unstructured environments. The proposed strategy combines an image segmentation algorithm with a dynamic planning algorithm and an occlusion algorithm, which increases the picking success rate of the harvester. The image segmentation algorithm (based on an SVM pixel classifier, a watershed transform and a point cloud registration) is responsible for detecting and localizing aubergines. Depending on the workspace, the locations of the fruits, and the arm configurations, the planning algorithm determines the movement sequence needed to grasp and detach the aubergines. These movements may involve either the simultaneous harvesting of two pieces of fruit or harvesting a single fruit with a single arm. Finally, the occlusion algorithm addresses aubergines that have low visibility due to leaf occlusions by planning a collaborative behavior between the arms to solve the occlusion and proceed with dual-arm harvesting. This cooperative operation mimics the complex human harvesting motion of using one arm to push leaves aside while the other arm picks the fruit.

The efficiency of the harvester was confirmed through laboratory tests. The experimental results show that the harvester can pick 91.67% of the total number of aubergines in the proposed common scenarios. Therefore, the robotic aubergine harvesting system shows a substantial level of validity. Moreover, we analyzed the failed scenarios and obtained interesting findings; for example, most of the failures were related to changing lighting conditions. Thus, future work to enhance the harvester robot should prioritize improvements to image acquisition.

REFERENCES

[1] Y. Edan, S. Han, and N. Kondo, *Springer Handbook of Automation*. Berlin, Germany: Springer, 2009, pp. 1095–1121.

[2] K. Kapach, E. Barnea, R. Mairon, Y. Edan, and O. Ben-Shahar, "Computer vision for fruit harvesting robots—State of the art and challenges ahead," *Int. J. Comput. Vis. Robot*, vol. 3, nos. 1–2, pp. 4–34, Apr. 2012.

[3] A. Gongal, S. Amatya, M. Karkee, Q. Zhang, and K. Lewis, "Sensors and systems for fruit detection and localization: A review," *Comput. Electron. Agricult.*, vol. 116, pp. 8–19, Aug. 2015.

[4] K. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, "Machine learning in agriculture: A review," *Sensors*, vol. 18, no. 8, p. 2674, Aug. 2018.

[5] H. Tian, T. Wang, Y. Liu, X. Qiao, and Y. Li, "Computer vision technology in agricultural automation—A review," *Inf. Process. Agricult.*, vol. 7, no. 1, pp. 1–19, Mar. 2020.

[6] A. Plebe and G. Grasso, "Localization of spherical fruits for robotic harvesting," *Mach. Vis. Appl.*, vol. 13, no. 2, pp. 70–79, Nov. 2001.

[7] S. Yamamoto, S. Hayashi, H. Yoshida, K. Kobayashi, and K. Shigematsu, "Development of an end effector for a strawberry-harvesting robot," in *Proc. Int. Conf. High Tech. Greenhouse Syst. Manage. (GREENSYS)*, Naples, Italy, 2008, pp. 565–572.

[8] Y. Onishi, T. Yoshida, H. Kurita, T. Fukao, H. Arihara, and A. Iwai, "An automated fruit harvesting robot by using deep learning," in *Proc. JSME Annu. Conf. Robot. Mechatronics (Robomec)*, 2018, pp. 1A1–B03.

[9] K. Tanigaki, T. Fujiura, A. Akase, and J. Imagawa, "Cherry-harvesting robot," *Comput. Electron. Agricult.*, vol. 63, no. 1, pp. 65–72, Aug. 2008.

[10] Y. Zhao, L. Gong, Y. Huang, and C. Liu, "A review of key techniques of vision-based control for harvesting robot," *Comput. Electron. Agricult.*, vol. 127, pp. 311–323, Sep. 2016.

[11] B. Zion, M. Mann, D. Levin, A. Shilo, D. Rubinstein, and I. Shmulevich, "Harvest-order planning for a multiarm robotic harvester," *Comput. Electron. Agricult.*, vol. 103, pp. 75–81, Apr. 2014.

[12] H. A. M. Williams, M. H. Jones, M. Nejati, M. J. Seabright, J. Bell, N. D. Penhall, J. J. Barnett, M. D. Duke, A. J. Scarfe, H. S. Ahn, J. Lim, and B. A. MacDonald, "Robotic kiwifruit harvesting using machine vision, convolutional neural networks, and robotic arms," *Biosyst. Eng.*, vol. 181, pp. 140–156, May 2019.

[13] X. Ling, Y. Zhao, L. Gong, C. Liu, and T. Wang, "Dual-arm cooperation and implementing for robotic harvesting tomato using binocular vision," *Robot. Auto. Syst.*, vol. 114, pp. 134–143, Apr. 2019.

[14] S. Hayashi, K. Ganno, Y. Ishii, and I. Tanaka, "Robotic harvesting system for eggplants," *Jpn. Agricult. Res. Quart.*, vol. 36, no. 3, pp. 163–168, 2002.

[15] (2019). *Eurostat—Agriculture*. [Online]. Available: https://ec.europa.eu/eurostat/home?

[16] I. W. I. Wan, W. H. Kit, and M. A. Awal, "Design and Development of Eggplant Harvester for Gantry System," *Pertanika J. Sci. Technol.*, vol. 18, no. 2, pp. 231–242, Jul. 2010.

[17] (2019). *Robotics Company | Robotic Assistive Technology | Kinova*. [Online]. Available: https://www.kinovarobotics.com

[18] A. Campeau-Lecours, H. Lamontagne, S. Latour, P. Fauteux, V. Maheu, F. Boucher, C. Deguire, and L.-J. C. L'Ecuyer, "Kinova modular robot arms for service robotics applications," *Int. J. Robot. Appl. Technol.*, vol. 5, no. 2, p. 23, 2017.

[19] E. Navas, R. Fernández, D. Sepúlveda, M. Armada, and P. Gonzalez-de-Santos, "Modular dual-arm robot for precision harvesting," in *Proc. Robot, 4th Iberian Robot. Conf.* (Advances in Intelligent Systems and Computing), Porto, Portugal. Cham, Switzerland: Springer, 2020.

[20] (2019). *MESA IMAGING SR4000/SR4500 User Manual Contents*. [Online]. Available: http://www.realtechsupport.org/UB/SR/range_finding/SR4000_SR4500_-Manual

[21] (2019). *Prosilica GC2450 | 5.0 Megapixel Sony ICX625 CCD Sensor—Allied Vision*. [Online]. Available: https://www.alliedvision.com/en/products/cameras/detail/ProsilicaGC

[22] D. Sepúlveda, R. Fernández, E. Navas, P. González-de-Santos, and M. Armada, "ROS framework for perception and dual-arm manipulation in unstructured environments," in *Proc. Robot, 4th Iberian Robot. Conf.* (Advances in Intelligent Systems and Computing), Porto, Portugal. Cham, Switzerland: Springer, 2020.

[23] D. Kaeli, P. Mistry, D. Schaa, and D. Zhang, "Case study: Image clustering," in *Heterogeneous Computing With OpenCL 2.0*. Burlington, MA, USA: Morgan Kaufmann, Jan. 2015, pp. 213–228.

[24] Y. Guo and A. S. Ashour, "Neutrosophic sets in dermoscopic medical image segmentation," in *Neutrosophic Set in Medical Image Analysis*. New York, NY, USA: Academic, 2019, pp. 229–243.

[25] J. Siswantoro, A. S. Prabuwono, A. Abdullah, and B. Idrus, "Automatic image segmentation using sobel operator and k-means clustering: A case study in volume measurement system for food products," in *Proc. Int. Conf. Sci. Inf. Technol. (ICSITech)*, Yogyakarta, Indonesia, Oct. 2015, pp. 13–18.

[26] S. B. Kotsiantis, "Supervised machine learning: A review of classification techniques," *Informatica*, vol. 31, no. 1, pp. 249–268, 2007.

[27] H. Zhu, F. Meng, J. Cai, and S. Lu, "Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation," *J. Vis. Commun. Image Represent.*, vol. 34, pp. 12–27, Jan. 2016.

[28] Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew, "A review of semantic segmentation using deep neural networks," *Int. J. Multimedia Inf. Retr.*, vol. 7, no. 2, pp. 87–93, Jun. 2018.

[29] Q. Geng, Z. Zhou, and X. Cao, "Survey of recent progress in semantic image segmentation with CNNs," *Sci. China Inf. Sci.*, vol. 61, no. 5, May 2018, Art. no. 051101.

[30] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, "A survey on deep learning techniques for image and video semantic segmentation," *Appl. Soft Comput.*, vol. 70, pp. 41–65, Sep. 2018.

[31] F. Lateef and Y. Ruichek, "Survey on semantic segmentation using deep learning techniques," *Neurocomputing*, vol. 338, pp. 321–348, Apr. 2019.

[32] H. Yu, Z. Yang, L. Tan, Y. Wang, W. Sun, M. Sun, and Y. Tang, "Methods and datasets on semantic segmentation: A review," *Neurocomputing*, vol. 304, pp. 82–103, Aug. 2018.

[33] Z. Lu, D. Chen, and D. Xue, "Survey of weakly supervised semantic segmentation methods," in *Proc. Chin. Control Decis. Conf. (CCDC)*, Shenyang, China, Jun. 2018, pp. 1176–1180.

[34] M. Zhang, Y. Zhou, J. Zhao, Y. Man, B. Liu, and R. Yao, "A survey of semi- and weakly supervised semantic segmentation of images," *Artif. Intell. Rev.*, Dec. 2019.

[35] R. Fernandez, H. Montes, J. Surdilovic, D. Surdilovic, P. Gonzalez-De-Santos, and M. Armada, "Automatic detection of field-grown cucumbers for robotic harvesting," *IEEE Access*, vol. 6, pp. 35512–35527, 2018.

[36] R. Nagalla, P. Pothuganti, and D. S. Pawar, "Analyzing gap acceptance behavior at unsignalized intersections using support vector machines, decision tree and random forests," *Procedia Comput. Sci.*, vol. 109, pp. 474–481, 2017.

[37] S. Beucher, "Watershed, hierarchical segmentation and waterfall algorithm," in *Mathematical Morphology and Its Applications to Image Processing*, J. Serra and P. Soille, Eds. Dordrecht, The Netherlands: Springer, 1994, pp. 69–76.

[38] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 6, pp. 583–598, Jun. 1991.

[39] H. Zhu, B. Zhang, A. Song, and W. Zhang, "An improved method to reduce over-segmentation of watershed transformation and its application in the contour extraction of brain image," in *Proc. 8th IEEE Int. Conf. Dependable, Autonomic Secure Comput.*, Dec. 2009, pp. 407–412.

[40] Z. Taylor. (2019). *Camera to Robotic Arm Calibration*. [Online]. Available: http://www.zjtaylor.com/

[41] (2019). *MoveIt Motion Planning Framework*. [Online]. Available: https://moveit.ros.org/

[42] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor, and S. Schaal, "STOMP: Stochastic trajectory optimization for motion planning," in *Proc. IEEE Int. Conf. Robot. Autom.*, Shanghai, China, May 2011, pp. 4569–4574.

[43] M. Stein, S. Bargoti, and J. Underwood, "Image based mango fruit detection, localisation and yield estimation using multiple view geometry," *Sensors*, vol. 16, no. 11, p. 1915, Nov. 2016.

[44] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "DeepFruits: A fruit detection system using deep neural networks," *Sensors*, vol. 16, no. 8, p. 1222, Aug. 2016.

[45] Y. Xiong, Y. Ge, L. Grimstad, and P. J. From, "An autonomous strawberry-harvesting robot: Design, development, integration, and field evaluation," *J. Field Robot.*, vol. 37, no. 2, pp. 202–224, Aug. 2019.

[46] Q. Feng, W. Zou, F. Fan, C. Zhang, and X. Wang, "Design and test of robotic harvesting system for cherry tomato," *Int. J. Agricult. Biol. Eng.*, vol. 11, no. 1, pp. 96–100, 2018.

[47] A. Mohamed, J. Shaw-Sutton, B. M. Green, W. Andrews, E. J. Rolley-Parnell, Y. Zhou, P. Zhou, X. Mao, M. P. Fuller, and M. F. Stoelen, "Soft manipulator robot for selective tomato harvesting," in *Proc. Precis. Agricult.*, Jul. 2019, pp. 799–805.
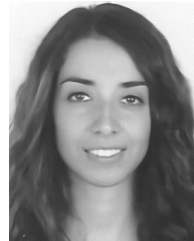
DELIA SEPÚLVEDA received the B.S. degree in telecommunications engineering and the M.Sc. degree in biomedical engineering from ETSIT, Polytechnic University of Madrid (UPM), in 2017 and 2018, respectively. From 2017 to 2018, she was a Researcher Assistant with the Robotics Laboratory, Photonic Technology and Bioengineering Department, ETSIT, UPM. She was working on the development of sensor networks for monitoring environments. In 2018, she did her internship with the National Hospital for Paraplegics, Spain. She is currently with the Centre for Automation and Robotics, CSIC, UPM. In the Biomechanics and Technical Aids Unit, she has participated with the Rehab Hand Project, developing virtual games with haptic feedback assistive robots for upper limb rehabilitation. She has published four articles in scientific journals and conference proceedings. Her research interests include control of human-robot-environment interaction, including robotic automation systems, study of improvements in control, and automation strategies.

ROEMI FERNÁNDEZ received the Ph.D. degree in industrial engineering from ETSII, Polytechnic University of Madrid (UPM), in 2006. She was a visiting Ph.D. Student with the Centre for Control Engineering and Computation, University of California, Santa Barbara, CA, USA, in 2003, and the Department of Automatic Control, Lund University, Sweden, in 2004. From 2005 to 2009, she was an Associate Professor with San Pablo CEU University. In 2008, she joined the Field and Service Robotics Group, Centre for Automation and Robotics, CSIC, UPM. Since 2009, she has also been collaborating as a Professor in master's program (automation and robotics) with ETSII, UPM, where she is currently a Researcher with the Centre for Automation and Robotics, CSIC. She has worked on several European Commission-Funded Projects, such as Crops (FP7-NMP-2009-LARGE-3.4-1 246252, a robotic platform for sustainable production, and harvesting of high-value crops), TIRAMISU (FP7-SEC-2011.4.2-2 284747, a toolbox implementation for removal of antipersonnel mines, submunitions, and unexploded ordnances), and the European Clearing House for Open Robotics Development (ECHORD, Experiment Call 2 – CATCH – Cucumber Gathering – Green Field Experiments). She is the author of ten patents. She has published over 70 articles in scientific journals and conference proceedings. Her research interests include design and control of field and service robotic systems, including the design of non-linear actuators, multisensory systems, and non-linear controllers. She served as an Expert Evaluator for the Research Executive Agency of the European Commission, an External Evaluator for the National Agency for Quality Assessment and Accreditation (ANECA), Spain, a Reviewer for several foreign research agencies, such as ISF, and a Reviewer for a large number of scientific journals and international conferences. She received the Award for Best Teaching, in 2008.

EDUARDO NAVAS received the B.S. degree in mechanical engineering from the University of Castilla-La Mancha, Spain, in 2017, and the M.Sc. degree in industrial engineering from the University of Valladolid, Spain, in 2019, where he is currently pursuing the Ph.D. degree in systems engineering, automation and robotics. From 2018 to 2019, he was a Researcher with the Social System Engineering Centre (INSISOC), and the Research Institute, University of Valladolid. He worked on a project about agent-based modeling, railway slot allocation, and combinatorial auctions – ABARNET. In 2019, he joined the Center for Automation and Robotics, CSIC, Polytechnic University of Madrid, as a Researcher, where he is actively involved in the design and development of dual-manipulators and field robotic systems. He is the author/coauthor of six articles published in scientific journals and conference proceedings. His main research interests include design of field robotic systems, soft robotics, and novel robotic applications in industry and service fields.

MANUEL ARMADA (Member, IEEE) received the Ph.D. degree in physics from the University of Valladolid, Spain, in 1979, and the Honored Doctor Honoris Causa from State Technical University, Moscow. Since 1976, he has been involved in research activities related to automatic control and has been in robotics, since 1979. He was the Director of the Instituto de Automatica Industrial (CSIC), from 2009 to 2010, and the Director of the Centre for Automation and Robotics, CSIC, Polytechnic University of Madrid, from 2012 to 2015. He is currently a Professor of research with CSIC. He has been a Principal Investigator in more than 50 RTD projects, including international ones, such as EUREKA, ESPRIT, BRITE/EURAM, GROWTH, ICT, NMP, INCO-COPERNICUS, and others outside EU, particularly in Latin America (CYTED-AECID) and Russia (ESSIDAC EC project and Joint Projects between RAS-CSIC). He has also carried out important work in several EC Thematic Networks, such as CLAWAR. He represents the Spanish Government with the International Advanced Robotics Programme (IARP). He has published over 50 articles in top scientific journals and more than 200 contributions to other journals and conferences. He holds over 20 international patents. His main research interests include robot design and control, with a special emphasis on force control and walking and climbing machines. He is a member of the IEEE RAS. He was the Chairman of the EC TELEMAN Evaluation Panel, a Reviewer for Foreign Research Agencies, and several EC FPs. He received the IMEKO TC17 Award and the CSIC Distinguished Award three times. He serves as the Topic Editor-in-Chief for the *International Journal of Advanced Robotic Systems*. He also serves as a reviewer for several other scientific journals.

PABLO GONZÁLEZ-DE-SANTOS received the B.E. degree in physics and the Ph.D. degree in automatic control from the University of Valladolid, Spain, in 1980 and 1986, respectively. In 1987, he joined the Institute of Industrial Automation, CSIC, as a Scientist. From 1990 to 1991, he was a Visiting Scientist with The Robotics Institute, Carnegie Mellon University, being involved with AMBLER Walking Machine Project, funded by NASA. In 2010, he joined the Centre for Automation and Robotics, CSIC, Polytechnic University of Madrid (UPM), Madrid, Spain, as the Director of the Department for Applied Robotics, continuing his activities in field robotics. He is currently a Research Professor with the Centre for Automation and Robotics-Association of the Spanish National Research Council (CSIC) and UPM. He has published approximately 70 articles indexed in the Journal Citation Reports (JCR) and a book on *Quadrupedal Locomotion*. He has edited five conference proceedings. Since 2010, he has been involved in robotics for agriculture. He has coordinated the European Commission FP7 Project Robot Fleets for Highly Effective Agriculture and Forestry Management (RHEA). He is focused on adapting the concept of a smart factory to the concept of a smart farm. His current research interests include where he was actively involved in the design and development of industrial manipulators, intelligent assistance devices, and service robots, specifically walking robots. He served as the Editor-in-Chief for the *International Journal of Advanced Robotic Systems*. He is a member of the Editorial Boards of the Journals *Industrial Robot* and *Advances in Robotics Research*.

• • •