agronomy

MDPI

Article

A Novel Segmentation Recognition Algorithm of *Agaricus bisporus* Based on Morphology and Iterative Marker-Controlled Watershed Transform

Chao Chen [1,2], Shanlin Yi [1,2], Jinyi Mao [1,2], Feng Wang [1,2], Baofeng Zhang [1,2] and Fuxin Du [3,4,*]

1 School of Mechanical Engineering, Yangzhou University, Yangzhou 225127, China
2 Jiangsu Engineering Center for Modern Agricultural Machinery and Agronomy Technology, Yangzhou 225127, China
3 School of Mechanical Engineering, Shandong University, Jinan 250061, China
4 Key Laboratory of High Efficiency and Clean Mechanical Manufacture of Ministry of Education, Shandong University, Jinan 250061, China
* Correspondence: dufuxin@sdu.edu.cn

Abstract: Accurate recognition of *Agaricus bisporus* is a prerequisite for precise automatic harvesting in a factory environment. Aimed at segmenting mushrooms adhering together from the complex background, this paper proposes a watershed-based segmentation recognition algorithm for *A. bisporus*. First, the foreground of *A. bisporus* is extracted via Otsu threshold segmentation and morphological operations. Then, a preliminary segmentation algorithm and a novel iterative marker generation method are proposed to prepare watershed markers. On this basis, a marker-controlled watershed algorithm is adopted to segment and recognize *A. bisporus* individuals. All the algorithms are implemented based on OpenCV (Open Source Computer Vision) libraries. Tests on images of *A. bisporus* collected at the cultivation bed show that the average correct recognition rate of the proposed algorithm is 95.7%, the average diameter measurement error is 1.15%, and the average coordinate deviation rate is 1.43%. The average processing time is 705.7 ms per single image, satisfying the real-time constraints based on 1 image/s. The proposed algorithm performed better than the current Circle Hough Transform (OpenCV's implementation). It is convenient and easy to operate, providing a sound basis for subsequent research on mechanized harvesting equipment for *A. bisporus*.

Keywords: *Agaricus bisporus*; segmentation recognition algorithm; computer vision; image processing; background filtering

check for updates

Citation: Chen, C.; Yi, S.; Mao, J.; Wang, F.; Zhang, B.; Du, F. A Novel Segmentation Recognition Algorithm of *Agaricus bisporus* Based on Morphology and Iterative Marker-Controlled Watershed Transform. *Agronomy* **2023**, *13*, 347. https://doi.org/10.3390/agronomy13020347

Academic Editors: Gamal Elmasry and Mario Cunha

Received: 15 November 2022
Revised: 25 December 2022
Accepted: 23 January 2023
Published: 25 January 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Agaricus bisporus is one of the world's most produced and consumed edible mushroom [1–3]. With the increasing demand for *A. bisporus*, its production mode has developed from traditional small-household cultivation to large-scale industrialized intelligent cultivation [4,5]. Several production processes have been mechanized, such as ruffling machines [6], continuous fill plate-put plate devices with second-phase compost [7], and intelligent control of the environment [8]. However, the picking process still relies on traditional manual labor. Like other agricultural industries, the mushroom industry is facing labor shortages and rising costs. The cost of manual picking is high, usually accounting for about 15–30% of total production costs [9]. In addition, there are other problems involving low picking efficiency, limited working time, and inconsistent grading standards. Mechanical harvesters have been used since the late 1970s and can significantly increase harvesting efficiency [9]. However, mechanical harvesting is not selective, and it is very harsh on the quality of the mushrooms [10]. Mushrooms can be bruised during mechanical harvesting. With the growth in scale and output of the *A. bisporus* industry, the development of a further intelligent automatic harvesting system for *A. bisporus* is necessary.

Accurate recognition of *A. bisporus* through computer vision is a prerequisite for intelligent harvesting. The medium of the *A. bisporus* seedling bed is complex and disturbed by weeds, soil, and a large number of mycelia. Additionally, mushrooms grow at irregular rates and intervals, causing irregular size, maturity, and spacing from each other, accompanied by tilt and adhesive overlap. Therefore, their recognition is very challenging. To this end, global scholars have conducted extensive research. Tillett and Batchelor [11] designed a recognition algorithm according to the different grayscale values of the top and edge of *A. bisporus.* However, it had high light requirements with a success rate of only 72%. Similarly, Yu et al. [12] segmented individual mushrooms based on a sequential scan algorithm according to the gray distribution characteristics of *A. bisporus* images. Qiu et al. [13] identified mushrooms by extracting mushroom boundaries, locating mushroom center coordinates, and calculating the perimeter. However, the effect of segmentation on overlapping mushrooms was limited. The above studies are based on RGB images for recognition. Ji et al. [14] introduced image depth information based on the three-dimensional longitudinal structure characteristics of *A. bisporus* and developed a "submersion method" to filter out the matrix background and segment the adherent mushroom clusters. Then, they used Hough circle detection to measure the diameter of *A. bisporus* and achieved a recognition success rate of 92.37%. Masoudian [15] segmented images in a more intuitive HSV color space and obtained each mushroom's center position and perimeter using the image gradient information and the Hough transform method.

All the above studies adopt traditional digital image processing techniques to identify targets based on color, shape, and texture features. With the development of artificial intelligence and computer computing power, some target recognition methods based on deep learning neural networks have been applied to recognize *A. bisporus*. Lee et al. [16] obtained a three-dimensional point cloud of *A. bisporus* with a depth camera and used a Faster regional convolutional neural network (R-CNN) model to segment overlapping and adherent mushroom clusters and identify individual mushrooms with an accuracy of 70.93%. Lu et al. [17,18] used a convolutional neural network to measure the mushroom size and to count the number of mushrooms. Cao [19] proposed a multimodal attention fusion network architecture based on the CenterNet target detection method and demonstrated through experiments that the recognition of *A. bisporus* can be better achieved under different light intensities. Deep learning methods can achieve higher target recognition rates than traditional digital image processing techniques. However, they require huge training datasets and long training time, and the algorithm runs for a long time with low real-time performance. Saddik et al. used embedded systems to perform complicated real-time tasks, such as monitoring indices, weed detection, and mapping agricultural soil [20–22]. Their work showed that the embedded system offers a low-cost solution to processing images in real-time and embedded systems based on artificial intelligence give more flexibility in the field of precision agriculture [23].

In summary, there are few studies on recognition algorithms for *A. bisporus*, and they have some application limitations. This paper thoroughly investigates the image information of *A. bisporus* and proposes a segmentation and recognition method. The main contributions of this paper are summarized as follows.

(1) We propose a foreground extraction method to extract *A. bisporus* from complex backgrounds.
(2) A preliminary segmentation algorithm is designed to solve the problem of pseudo-extreme points formed after distance transformation for large-scale adherent mushroom clusters.
(3) An iterative marker generation method is proposed to effectively mark individual mushrooms from adherent mushroom clusters by dynamically adjusting the threshold value of the global threshold segmentation algorithm, together with distance transformation and area filtering. Then, we use the marker-controlled watershed algorithm to identify and segment mature mushroom individuals.

2. Materials and Methods

The flow of the *A. bisporus* segmentation and recognition algorithm is shown in Figure 1. First, the image background was filtered using histogram equalization [24,25], Otsu threshold segmentation [26], and morphological operations [27,28] to extract foreground images containing only *A. bisporus.* Then, individual mushroom markers were prepared by preliminary segmentation as well as by the iterative marker generation method based on distance transformation, global threshold segmentation, and area filtering. The *A. bisporus* was subsequently segmented using a marker-controlled watershed algorithm. Finally, the bounding rectangle method was adopted to calculate mushroom cap diameter and center position to identify and locate *A. bisporus.* All algorithms were implemented based on OpenCV (Open Source Computer Vision) libraries, which include a wide catalog of highly optimized computer vision algorithms.

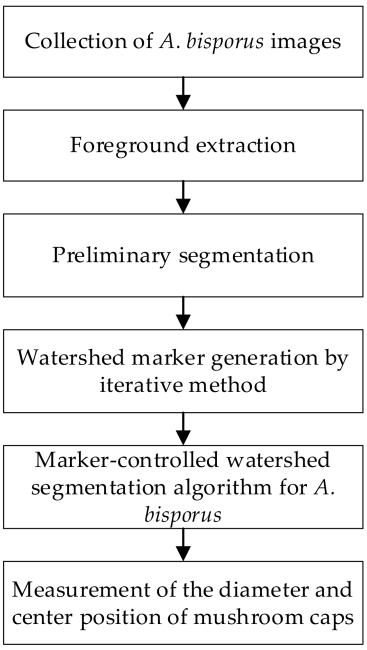


Figure 1. Flow chart of *A. bisporus* segmentation and recognition algorithm.

2.1. Background Filtering and Foreground Extraction

The size of the image to be processed was 600 × 450 pixels. After acquiring the grayscale image of *A. bisporus*, the contrast of the *A. bisporus* image was first enhanced using the histogram equalization algorithm. Then the image was binarized using the Otsu algorithm, as shown in Figure 2. The dark matrix background and the light mushroom foreground in the *A. bisporus* image should be easily distinguishable after binarization. However, the white mycelia and immature mushrooms across the matrix blur the boundary between the matrix and the mature mushroom, causing significant interference in identifying *A. bisporus*.

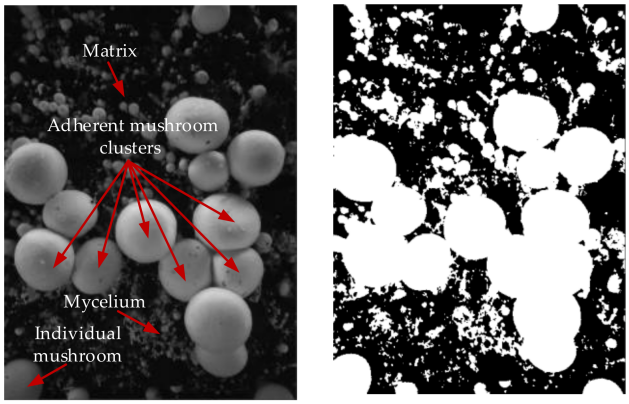


Figure 2. Images of *A. bisporus*. (**a**) Original image; (**b**) Binary image (Otsu algorithm).

For the effective recognition and localization of *A. bisporus*, the cluttered matrix background should first be filtered to obtain the foreground image of *A. bisporus*. The process is shown in Figure 3 and Algorithm 1.

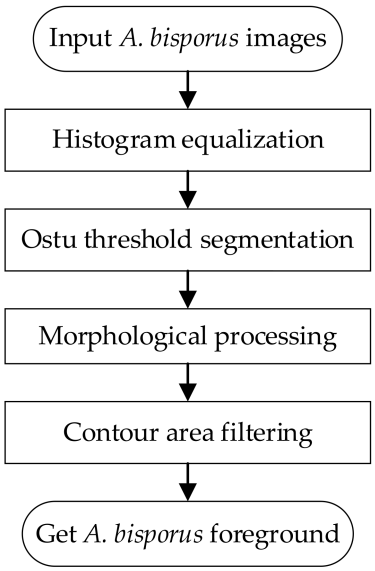
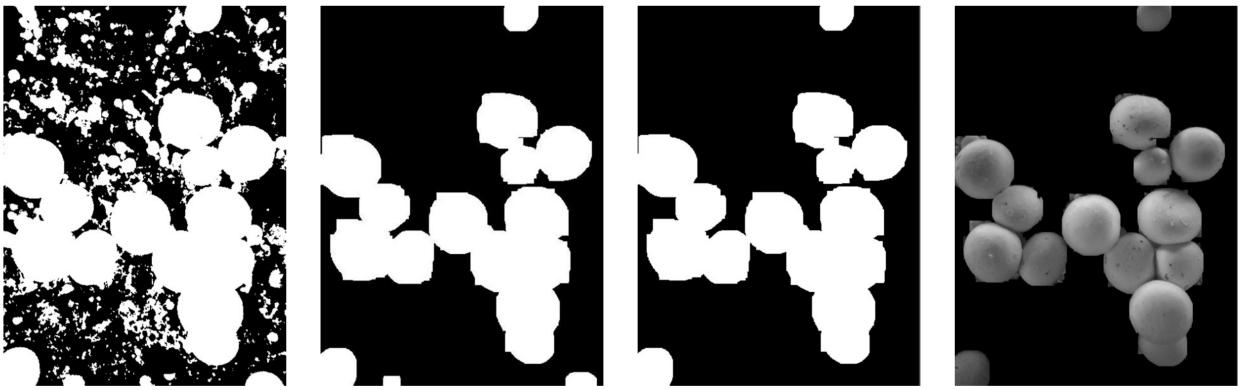


Figure 3. The background filtering and foreground extraction process of *A. bisporus* images.

As shown in Figure 2, the domains corresponding to mature *A. bisporus* after binarization are large white blocks, and scattered mycelia and immature mushrooms form many small noises, burrs, and fine connections in the matrix background. The binary image was processed using the morphological opening operation according to the image features. After comparing several trials, a structural element with a rectangular shape and 7×7 size was finally used for multiple open operations, considering the real-time demand of image recognition. The ratio of pixel size of the structural element to pixel size of the image is 1.81%. The results are shown in Figure 4b. As shown in the figure, the mycelia and small immature mushrooms in the background have been filtered out, but not the mushrooms at the edges of the image that were not fully captured due to the limitation of camera views (collection completeness was less than 50%). There can be a significant error in positioning and measuring the diameter of these mushroom caps. Therefore, the area of all contours in Figure 4b was calculated using the *contourArea* function [29], and the contours of small areas corresponding to incomplete mushrooms were filtered out, thereby obtaining Figure 4c with the background completely filtered. The original image was processed with the "and" operation to obtain the foreground image of *A. bisporus*.

Algorithm 1 Background filtering and foreground extraction.

Start algorithm
Inputs: *A. bisporus* images
Function Enhance image contrast
    Obtains an image grayscale histogram
    Histogram equalization
End Function
Function Image binarization
    Otsu finds the optimal threshold T
    IF The pixel grayscale value is higher than T ***Then***
        The pixel grayscale value is assigned 255
    End IF
    Else The pixel grayscale value is less than or equal to T ***Then***
        The pixel grayscale value is assigned 0
    End Else
End Function
Function Morphological processing
    Structural element erodes binary image
    Structural element dilates binary image
End Function
Function Contour area filtering
    Calculate the area of each region
    Remove small regions
End Function
End algorithm
Output: The foreground of *A. bisporus* images

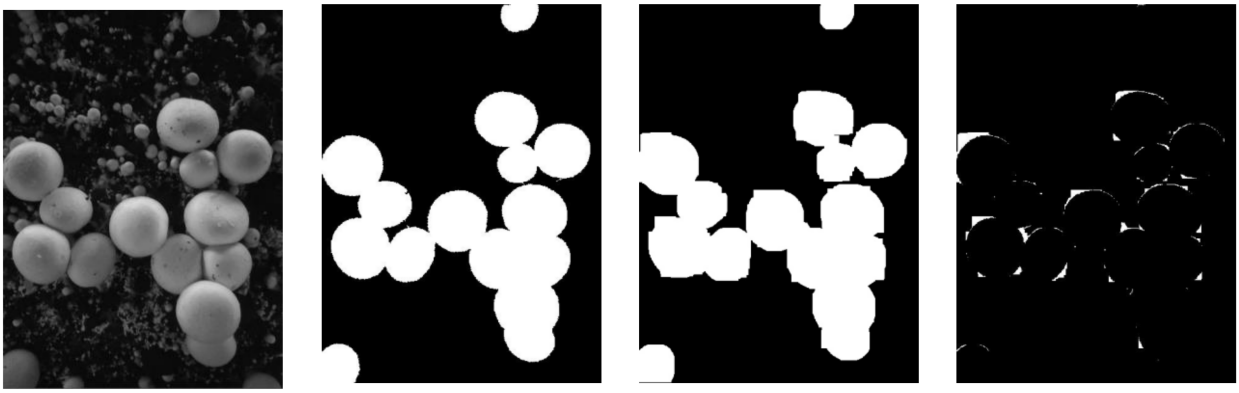


(**a**) (**b**) (**c**) (**d**)

Figure 4. Background filtering and foreground extraction of *A. bisporus*. (**a**) Binarized image; (**b**) Morphological processing; (**c**) Contour area filtering; (**d**) "And" operation on the grayscale image.

To verify the foreground extraction effect of the proposed algorithm, we manually separated the *A. bisporus* region in the image using Photoshop software, and the separation result was used as the standard image. The error between the extraction results using the proposed algorithm and the standard image was calculated, as shown in Figure 5. Here, Figure 5a is the grayscale image of *A. bisporus*; Figure 5b shows the manual extraction result; Figure 5c indicates the result of the foreground extraction algorithm presented in

this paper; Figure 5d, the error binary image, shows the difference between Figure 5b,c, representing the background parts misused as foreground.



(**a**) (**b**) (**c**) (**d**)

Figure 5. Comparison of foreground extraction algorithm results and manual extraction results. (**a**) Grayscale image; (**b**) Manual extraction results; (**c**) Background filtering results; (**d**) Error binary image.

To objectively measure the algorithm's foreground extraction performance, we introduce the error rate (*ER*) used in the literature [30] as the objective evaluation criterion. The *ER* is a common method to evaluate the quality of image segmentation and is calculated as follows:

$$ER = \frac{N_e}{N_s} \times 100\% \quad (1)$$

where N_e denotes the total number of pixels in the error region, i.e., the total number of pixels in the white region in Figure 5d, and N_s represents the total number of pixels in the image. The proposed algorithm was used to process three sets of *A. bisporus* images.

To find the impact of resolution, we compared the performance of the algorithm with the same image at different resolutions. The results of the ER are shown in Table 1, indicating that the interference factors in the matrix of *A. bisporus* can be basically removed, laying a good foundation for the subsequent recognition of *A. bisporus*. From the table, we can see that the resolution has minimal impact on the ER. We subsequently chose low-resolution images to implement the segmentation algorithm to reduce processing time.

Table 1. The ER of the foreground extraction of different *A. bisporus* images.

Image Serial Number	Image Size (Pixels × Pixels)	Pixel Number in the Error Area N_e (Pixels)	Error Rate (ER) (%)
1	600 × 450	6535	2.42
	1200 × 900	27,101	2.51
	3072 × 2048	153,748	2.44
2	600 × 450	4324	1.60
	1200 × 900	18,576	1.72
	3072 × 2048	103,809	1.65
3	600 × 450	2678	0.99
	1200 × 900	12,253	1.13
	3072 × 2048	65,714	1.04

2.2. Watershed Algorithm Segmentation Based on Iterative Markers

For intelligent picking of *A. bisporus*, it is essential to obtain the diameter and center position of the cap of each individual mushroom. Therefore, the algorithm does not only

need to extract the foreground of *A. bisporus* but also to segment and identify the adherent mushrooms accurately.

2.2.1. Principle of Watershed Algorithm Based on Distance Transformation

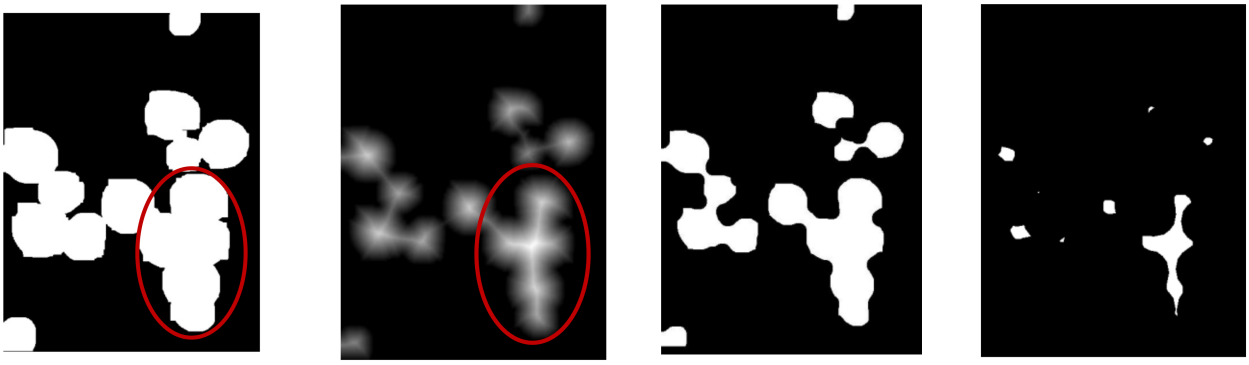
The watershed algorithm is a widely-used image segmentation method based on geomorphic analysis [31,32]. It analogizes the grayscale space of images to the geographic structure of the earth's surface, uses local gradient difference in touching areas, and identifies dividing lines through a simulation process of water surface rising from kernel areas under the pixel gradient constraints.

The surface grayscale values of mature *A. bisporus* are similar, which is suitable for the watershed algorithm. However, the traditional watershed algorithm is susceptible to noise points and other disturbances and produces over-segmentation due to excessive local minima. These problems can be improved by manual markers. When segmenting the image, a marker-controlled watershed algorithm designates the pixels with relatively high grayscale values as the starting points of the water rise, i.e., "marking" [33,34]. Ideally, each target *A. bisporus* to be segmented will correspond to a marker. Too many markers will lead to over-segmentation and an increased false rate, while too few markers will cause some mushrooms to be unrecognizable and an increased miss rate. We chose the distance transformation algorithm [35] to obtain more accurate foreground markers. The distance transform can convert an input binary image into a grayscale image. The grayscale value of each pixel is related to its closest distance to the edge of the background. A farther distance indicates greater pixel grayscale values. Euclidean distance (DIST_L2) [36] is used as the distance measure for transformation in this algorithm. The distance calculation formula is as follows:

$$\rho = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2)$$

where ρ is the Euclidean distance between pixel points (x_1, y_1) and (x_2, y_2).

The foreground image of *A. bisporus* obtained in the previous section was converted into a binary image. Then, a foreground marker image was obtained after Otsu binarization, distance transformation, and global threshold segmentation, as shown in Figure 6.



(**a**) (**b**) (**c**) (**d**)

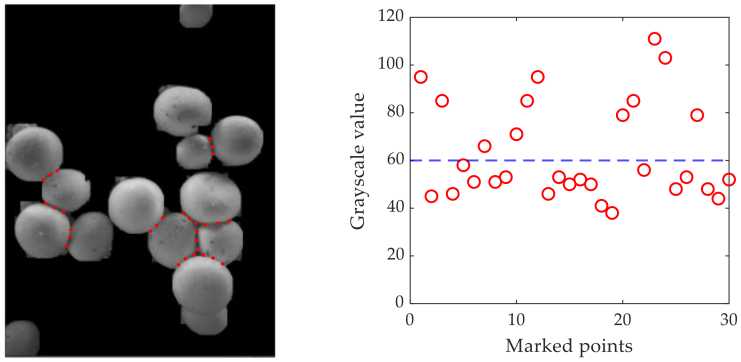
Figure 6. Effects of watershed markers prepared by distance transformation and global threshold segmentation. (**a**) Foreground binary image; (**b**) Distance transformation; (**c**) Global threshold segmentation after distance transformation (low threshold); (**d**) Global threshold segmentation after distance transformation (high threshold).

As shown in the figure, the effect of directly using the distance transformation algorithm to prepare markers is not satisfactory. There are two reasons: One is that large-scale adhesion of mushrooms can lead to pseudo-extreme points, resulting in false segmentation, as shown in the red circles in Figure 6. The other is that the diameter of each mushroom is different. After distance transformation of mushrooms with significant differences in diameter, the grayscale value difference at the extreme value points in the center of the

mushroom caps is also significant. If the threshold value is too low for global threshold segmentation, adhering mushrooms cannot be segmented, as shown in Figure 6c. If the threshold value is too high, the mushrooms with low central grayscale values will be ignored, resulting in missed picking, as shown in Figure 6d. To obtain accurate segmentation results, we must solve the above two problems.

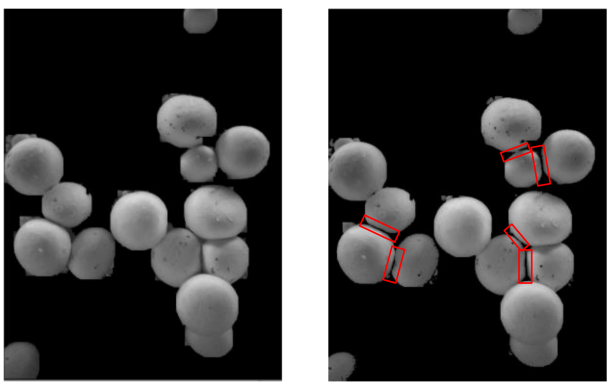
2.2.2. Preliminary Segmentation of Adherent Mushroom Clusters

To solve the issue of pseudo-extreme points, we preliminarily segmented the adherent mushroom clusters to reduce the adhesion area. Observing the foreground image of *A. bisporus*, we find that the color at the contact border of the adherent mushrooms is darker than at the edge of the mushroom cap, i.e., the grayscale value of the pixel at the border is lower than that of the pixel at the edge of the mushroom cap. Adherent mushroom clusters can be preliminarily segmented according to this feature. The threshold is set according to the grayscale value of the border and the mushroom cap edge, so that the threshold is slightly higher than the grayscale value of the border and lower than that of the mushroom cap edge. In looping through the foreground grayscale image of *A. bisporus*, pixels with grayscale values below this threshold are assigned 0 to achieve preliminary segmentation. To determine the preliminary segmentation threshold, we marked the ends and midpoints of the mushroom edge adhesions at 30 points and measured the grayscale values of each marked point. The results are shown in Figure 7. The grayscale values of 19 marked points were less than 60, so we set the preliminary segmentation threshold as 60. A higher threshold would cause holes in the mushroom cap area. The preliminary segmentation effect is shown in Figure 8.



(**a**) (**b**)

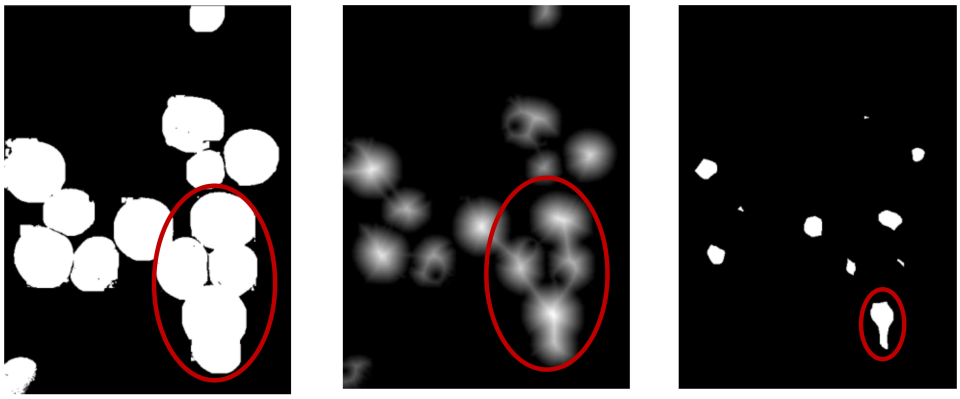
Figure 7. Determination of preliminary segmentation threshold. (**a**) Marked points; (**b**) Grayscale values of marked points.



(**a**) (**b**)

Figure 8. Comparison of preliminary segmentation effects. (**a**) Before preliminary segmentation; (**b**) After preliminary segmentation.

As shown in Figure 8, the adherent mushrooms with clear contact boundaries have been preliminarily separated. It should be noted that if a higher segmentation threshold is set, more adherent mushrooms can be segmented. However, the color of the *A. bisporus* cap edge and its center is inconsistent. The color of its center is slightly yellowish, and its pixel grayscale value is lower than that of the edges. If the threshold value for the preliminary segmentation is set too high, the pixel grayscale value of the mushroom cap center will also be set to 0, which would cause holes. In addition, the shadows caused by uneven lighting may also be mistaken as boundaries. Therefore, the preliminary segmentation threshold should be chosen at a reasonable level. The purpose of preliminary segmentation is to solve the problem of pseudo-extreme points formed after distance transformation for large-scale adherent mushroom clusters. Even if only part of the adherent mushrooms is segmented, the distance transformation results will be improved. The distance transformation was performed on the preliminarily segmented image, and a high threshold was selected for the global threshold segmentation of the transformation result to prepare the watershed markers. The results are shown in Figure 9.



(**a**) (**b**) (**c**)

Figure 9. Effects of watershed markers prepared by distance transformation and global threshold segmentation after preliminary segmentation. (**a**) Foreground binary image; (**b**) Distance transformation after preliminary segmentation; (**c**) Segmentation results of high thresholds.

By comparing with Figure 6, it can be seen that the problem of pseudo-extreme points formed after distance transformation for large-scale adherent mushroom clusters has been effectively solved. However, there remains the problem of global threshold segmentation under scale difference. The mushrooms in the circle in Figure 9c require a higher threshold to be segmented, but the current threshold is already high enough to filter out multiple small mushrooms. In the case of such scale differences, how to reasonably choose the global segmentation threshold to generate the corresponding marker for each mushroom is the next problem to be solved.

2.2.3. Watershed Segmentation Algorithm Based on Iterative Markers

Aimed at resolving the problem of global threshold segmentation under scale differences, this paper proposes an iterative threshold segmentation method combined with area filtering. More specifically, the idea is to first select a low threshold to segment the image obtained from distance transformation. The markers obtained can be divided into segmented mushroom individuals and adherent mushrooms that need further segmentation. The marker areas corresponding to mushroom clusters is larger than those corresponding to individuals. Thus, the individual mushroom markers can be retained through contour area filtering, and the markers corresponding to the adherent mushrooms can be separately used as a new input image for another distance transformation and global segmentation algorithm with a higher threshold value to segment again. The iteration is performed until the contour area of the markers obtained by segmentation is not greater than a certain small area value. This area value represents the maximum contour area of the markers

corresponding to individual mushrooms and can be obtained statistically by experiment. When the marker contour area obtained from the global threshold segmentation is smaller than this area value, it is considered that all adherent mushrooms have been successfully segmented and marked. Taking Figure 9a as the example, obtained by binarization after preliminary segmentation, the process of preparing the watershed markers by this iterative method is shown in Figure 10.

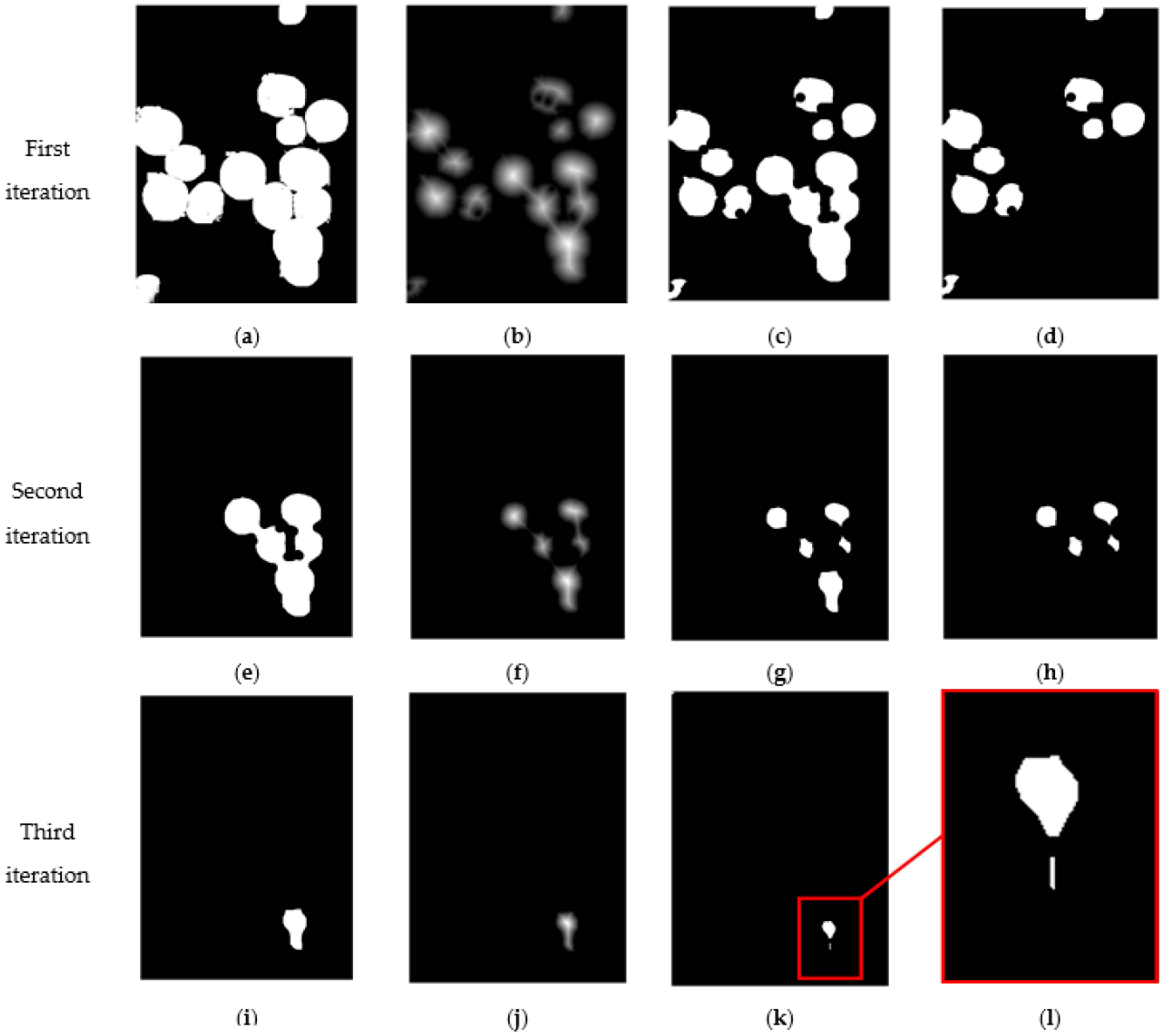


Figure 10. Process of threshold iteration to prepare watershed markers. (**a**) Input image; (**b**) Distance transformation; (**c**) Global threshold segmentation; (**d**) Contour area filtering; (**e**) Input contour image of adherent mushrooms; (**f**) Distance transformation; (**g**) Global threshold segmentation; (**h**) Contour area filtering; (**i**) Input contour image of adherent mushrooms; (**j**) Distance transformation; (**k**) Global threshold segmentation; (**l**) Enlarged view of local contour.

The individual mushroom markers filtered after each iteration were combined to obtain a complete *A. bisporus* foreground marker image corresponding to individual mushrooms, as shown in Figure 11.

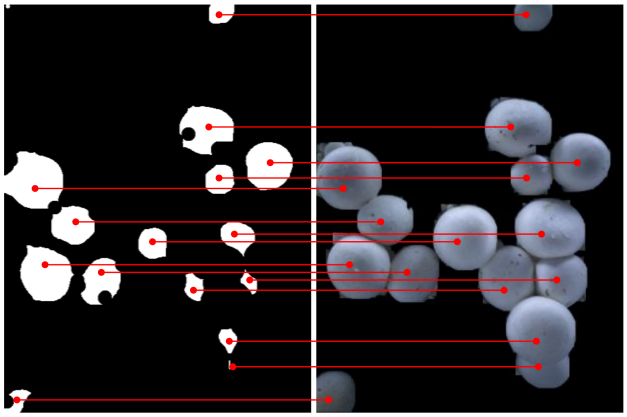
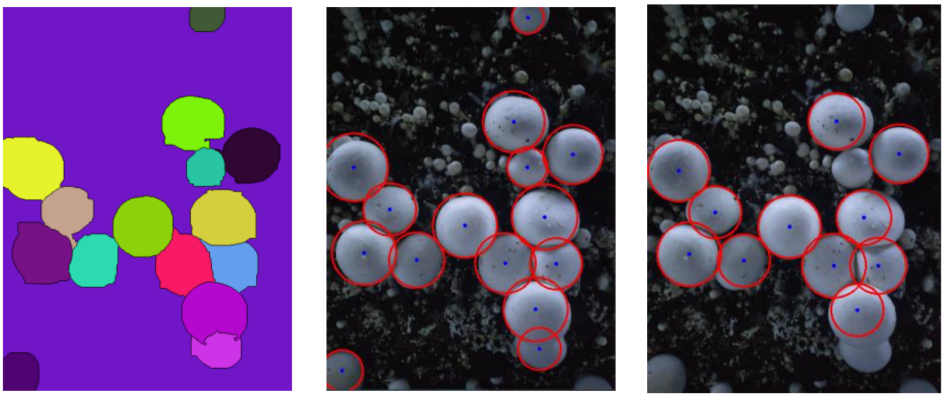


Figure 11. Complete watershed maker image and corresponding foreground image of *A. bisporus*.

After the complete *A. bisporus* marker image was created, the marker-controlled watershed algorithm created seed growth points according to the marker contours to divide similar regions, thus achieving complete segmentation of *A. bisporus*, as shown in Figure 12a. After the segmentation, the minimum bounding rectangle function was adopted to count each region's center and side length, which were used as the basis for the subsequent intelligent grasping algorithm. Taking the center of the minimum bounding rectangle of each region as the center of a circle and the long side as the diameter, we marked all of the recognized mushrooms in the original image, as shown in Figure 12b. Meanwhile, we compared results obtained by Circle Hough Transform (CHT; provided by the OpenCV library) to verify the proposed algorithm's performance. The five parameters of the OpenCV *HoughCircles* function were adjusted to 30, 100, 20, 30, and 50, and the CHT results are shown in Figure 12c.



(**a**) (**b**) (**c**)

Figure 12. Results of image processing. (**a**) Watershed segmentation results; (**b**) Mushroom detection by the proposed algorithm; (**c**) Mushroom detection by CHT.

We can see from Figure 12 that the proposed algorithm recognized all of the mushrooms. In contrast, CHT only recognized 11 of the 15 mushrooms. Moreover, some mushroom diameters and center coordinates identified by CHT had significant errors. We will present a quantitative comparison of the two algorithms in Section 3.

The complete flow of the watershed algorithm with iterative markers to segment *A. bisporus* is shown in Figure 13.

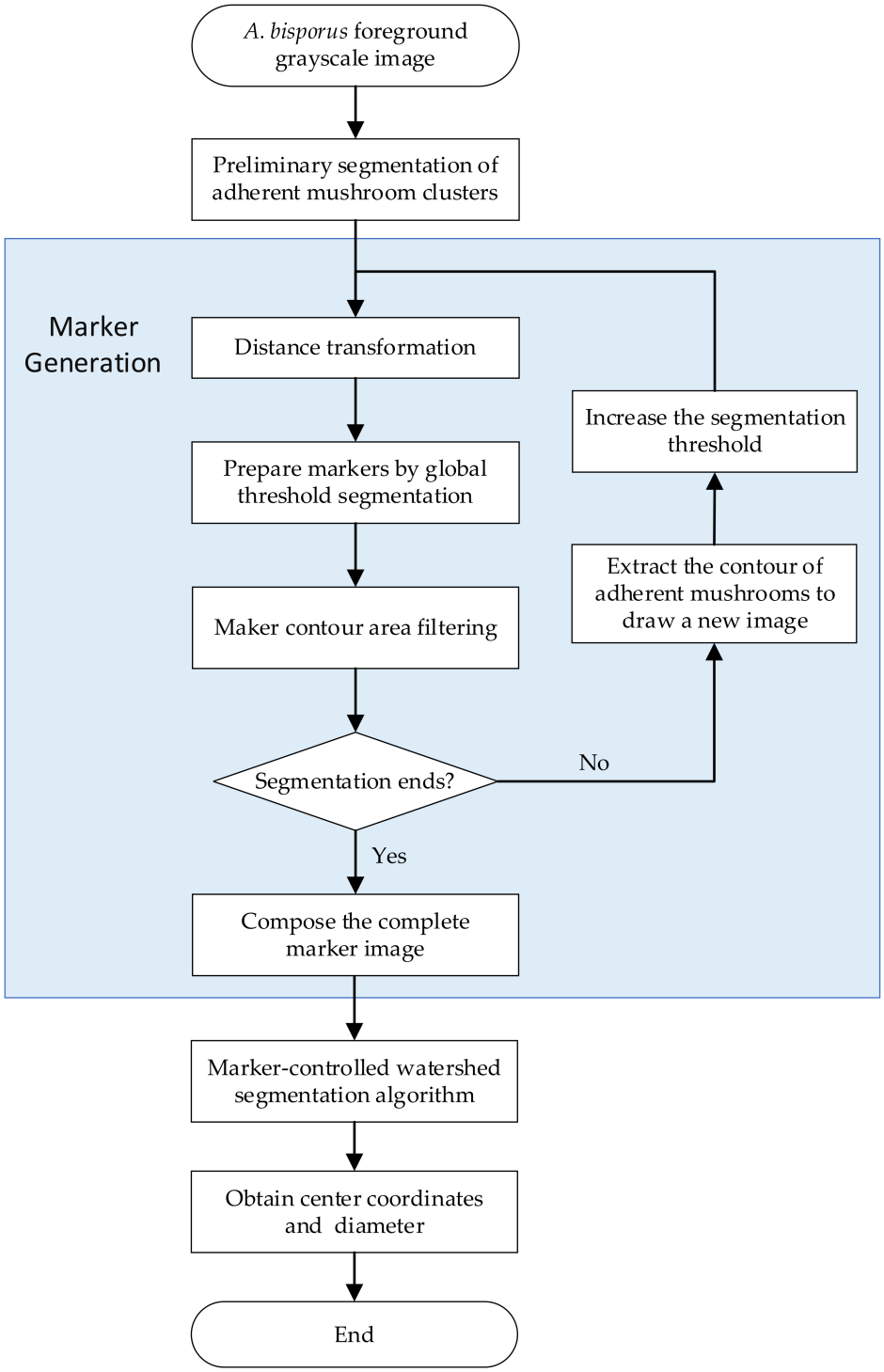


Figure 13. Flow chart of the watershed algorithm with iterative markers to segment *A. bisporus*.

As shown in the diagram, the segmentation algorithm consists of four processes: preliminary segmentation, marker generation by an iterative method, marker-controlled watershed segmentation, and acquisition of mushroom center coordinates and diameter. First, the adherent mushroom clusters were preliminarily separated by means of pixel manipulations (see Section 2.2.2). Second, a novel iterative method was applied to generate the watershed markers (see Section 2.2.3 and Figure 10). The result was that each mushroom was assigned a marker. Then, the marker-controlled watershed algorithm was employed to segment the adherent mushrooms and recognize the mushroom contours. Finally, the

minimum bounding rectangle function was adopted to determine the center coordinates and diameters of the mushrooms.

3. Experiment and Results Analysis

To verify the algorithm's effectiveness, we collected images for testing at the modern agricultural science and technology experimental field in Jiangyan District, Taizhou City, China.

3.1. System Hardware and Software

The Hikvision MV-CA060-10GC industrial camera (Hikvision Inc., Hangzhou, Zhejiang Province, China), with a resolution of 3072 × 2048, was used to obtain images of *A. bisporus*. The lens model was Hikvision MVL-HF0628M-6MPE, with a focal length of 6 mm and aperture set to F2.8. The camera was installed on a mobile gantry platform with a strip of white LED lights on both sides of the camera. The camera was placed perpendicularly from the soil surface, and the distance between the camera and the soil surface was 300 mm. The camera was set to auto white balance mode and maintained maximum light intake when the system was operated, and manual focus was used to obtain clear images.

The *A. bisporus* recognition algorithm was developed based on OpenCV 4.1.0 and Visual Studio 2015. The algorithm was implemented on an industrial personal computer with Intel(R) Core (TM) i7-6700HQ 2.6GHz CPU, 8.00GB RAM and Windows 10 operating system (OS). The Kithara RealTime Suite (KRTS), a modular real-time extension for Windows OS, was installed to realize real-time image processing with OpenCV.

3.2. Target Detection

We captured ten mushroom images in random places on the cultivation bed. To improve the efficiency of the algorithm, we resized the image to 600 × 450 with the *resize* function of OpenCV. Then, the images were processed using the proposed algorithm and CHT for comparison.

The tests focused on correct detection rate, miss rate, false rate, $F-Score$, and detection time. Their calculation formulae are as follows [37]:

$$DR = \frac{TP}{TP + FN} \tag{3}$$

$$MD = \frac{FN}{TP + FN} \tag{4}$$

$$FD = \frac{FP}{TP + FP} \tag{5}$$

$$F - Score = \frac{2 * Precision * DR}{Precision + DR},\ Precision = \frac{TP}{TP + FP} \tag{6}$$

where DR is the correct detection rate; MD is the miss rate; FD is the false rate; TP denotes true positive, meaning the algorithm successfully recognizes the target mushroom in the image; FN denotes false negative, meaning the algorithm recognizes a mushroom in the image as the background; and FP denotes false positive, meaning the algorithm recognizes the background area as a mushroom. The highest possible $F-Score$ is 1.0, indicating perfect precision and recall.

3.2.1. Test Results of the Proposed Algorithm

The test results of the proposed algorithm are shown in Table 2.

Table 2. Experimental results of the proposed algorithm.

Image Serial Number	Total Number of *A. bisporus*	Number of Correct Detections	Correct Rate	Number of Missed Detections	Miss Rate	Number of False Detections	False Rate	Detection Time (ms)
1	21	21	100%	0	0	0	0	752
2	11	10	91%	1	9%	0	0	692
3	15	15	100%	0	0	0	0	728
4	10	10	100%	0	0	0	0	680
5	7	7	100%	0	0	0	0	673
6	16	14	93.3%	1	6.7%	1	6.7	755
7	7	6	85.7%	1	14.3%	0	0	652
8	11	11	100%	0	0	0	0	703
9	10	9	90%	1	10%	0	0	696
10	9	8	89%	1	11%	0	0	726
Total	117	111	95.7%	5	4.3%	1	0.89%	705.7

As shown in Table 2, the average detection rate of the ten groups of tests was 95.7%, which is higher than those of 70.93% [16] and 92.37% [14] found in the literature. The $F-Score$ value was 0.97, which indicates that the algorithm has good precision and recall. Therefore, the algorithm accuracy met the actual picking requirements. Figure 14 shows the time spent by the algorithm on each image. The running time of a single image was between 652 ms and 755 ms, with an average time of 705.7 ms, which is sufficient to satisfy the real-time constraints based on 1 image/s. The time spent had an apparent upward trend with increases in the number of mushrooms. Nevertheless, the overall increase was small. We analyzed the images represented by the two outliers and discovered that the higher time consumption could be attributed to more complicated backgrounds, which required more time in the background filtering.

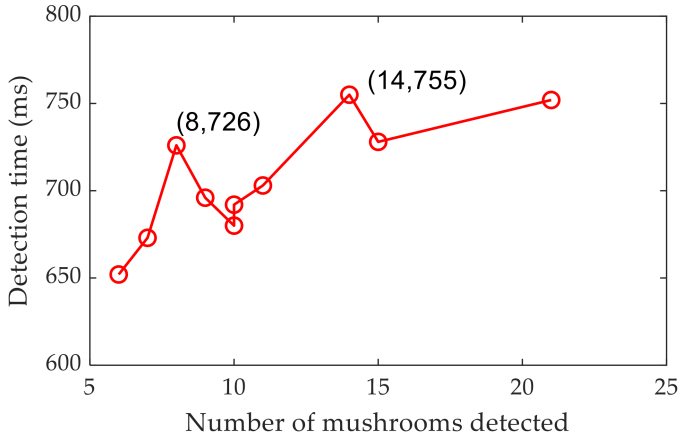
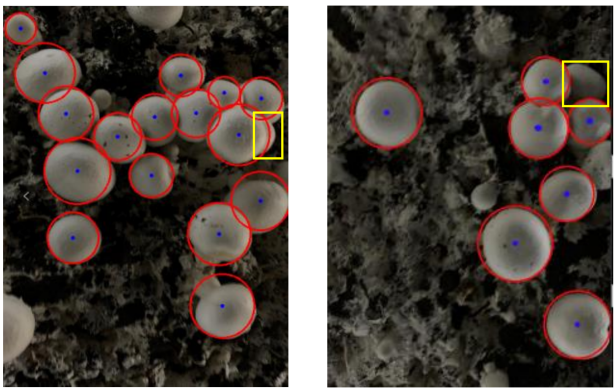


Figure 14. Detection time of images with different numbers of mushrooms.

Table 2 shows that the average miss rate of the algorithm was 4.3%, and the false rate was 0.89%. There are two causes for missed detections: One is that the height of individual *A. bisporus* can vary significantly at the adhesion points, and the lower individual *A. bisporus* can be almost completely obscured, as shown in Figure 15a. The other is that uneven light can lead to low grayscale values for some *A. bisporus*, which are mistaken for the background and missed, as shown in Figure 15b. The main reason for false detection is that the shape and grayscale value of some mycelium areas are close to the performance of *A. bisporus*, so they are mistaken as individual *A. bisporus*. The probability of false detection is extremely low due to the excellent effect of the algorithm in extracting the foreground.



(**a**) (**b**)

Figure 15. Analysis of missed detections. (**a**) Non-detection due to obscured lower mushrooms; (**b**) Non-detection due to uneven lighting.

The analysis shows that lighting significantly influences the false and missed detection rates. It is necessary to design more appropriate light sources to further optimize the light environment. The stalk of *A. bisporus* has a significant impact on the false rate. Therefore, more accurate segmentation techniques to separate the stalk from the mushroom cap can be explored in future studies to reduce the false rate.

3.2.2. Comparison between the Proposed Algorithm and CHT

The CHT test results are shown in Table 3.

Table 3. Experimental results of CHT.

Image Serial Number	Total Number of *A. bisporus*	Number of Correct Detections	Correct Rate	Number of Missed Detections	Miss Rate	Number of False Detections	False Rate	Detection Time (ms)
1	21	17	81.0%	4	19.0%	0	0	172
2	11	7	77.8%	2	22.2%	2	22.2%	151
3	15	11	73.3%	4	26.7%	0	0	197
4	10	10	100%	0	0	0	0	169
5	7	6	85.7%	1	14.3%	0	0	163
6	16	15	93.8%	1	6.2%	0	0	176
7	7	6	85.7%	0	0	1	14.3%	155
8	11	6	54.5%	5	45.5%	0	0	152
9	10	7	70.0%	3	30.0%	0	0	172
10	9	7	87.5%	1	12.5%	1	12.5%	146
Total	117	92	81.4%	21	18.6%	4	4.2%	165.3

Table 3 shows that the average detection rate of CHT is 81.4%, the average miss rate is 18.6%, and the false rate is 4.2%. The calculated $F - Score$ is 0.88. Comparing Tables 2 and 3, it can be seen that the correct detection rate of the proposed algorithm is higher than that of CHT. In addition, the proposed algorithm has lower miss and false rates. When the mushroom image is not close to a perfect circle or the grayscale value of the mushroom is low, the CHT algorithm is prone to miss or falsely detect them. From the experimental results, the proposed algorithm performed better than CHT in recognizing mushrooms.

3.3. Verification of Diameter Measurement and Center Coordinates

3.3.1. Verification of Diameter Measurement

In this paper, we use spatial resolution to calculate the actual diameter (measured in millimeters) after extracting the mushroom's diameter (measured in pixels) from the image. Spatial resolution [18] represents the number of independent pixel values per millimeter and is 2.1 pixel/mm in this paper. The mushrooms in Figure 2 were processed to verify diameter measurement. It should be noted that the diameters we measured are

the maximum diameters of the mushrooms. The relative error of the measurement results is calculated as follows:

$$ME = \frac{|EV - AV|}{AV} \times 100\% \quad (7)$$

where ME represents the measurement error, AV denotes the actual value of diameter, and EV denotes the experimental value measured by an algorithm. The results of the proposed algorithm and CHT are shown in Tables 4 and 5, respectively.

Table 4. Results of diameter measurement by proposed algorithm.

NO.	**1**	**2**	**3**	**4**	**5**
AV (mm)	23.7	46.6	29.9	44.1	51.8
EV (mm)	23	46.5	30.2	44.5	51.4
ME (%)	2.95	**0.21**	1	0.91	0.77
NO.	6	7	8	9	10
AV (mm)	39.4	51.8	37	43.2	47.5
EV (mm)	39.8	51.3	36.6	43.4	48
ME (%)	1.02	0.97	1.08	0.46	1.05
NO.	11	12	13	14	15
AV (mm)	39.3	46.1	41.3	43.7	28.4
ME (mm)	39.4	46.4	39.9	44.5	28.6
ME (%)	0.25	0.65	**3.39**	1.83	0.70
Average ME: **1.15%**					

Table 5. Results of diameter measurement by CHT.

NO.	**1**	**2**	**3**	**4**	**5**
AV (mm)	23.7	46.6	29.9	44.1	51.8
EV (mm)	miss	41	miss	42.9	40
ME (%)		11.83		2.72	22.78
NO.	6	7	8	9	10
AV (mm)	39.4	51.8	37	43.2	47.5
EV (mm)	41	38.1	miss	46.7	47.1
ME (%)	4.06	**26.45**		8.10	**0.84**
NO.	11	12	13	14	15
AV (mm)	39.3	46.1	41.3	43.7	28.4
ME (mm)	38.1	46.7	38.1	44.8	miss
ME (%)	3.05	1.30	7.75	2.52	
Average ME: **8.31%**					

As shown in Table 4, the maximum measurement error of the proposed algorithm was 3.39% (in bold), the minimum error was 0.21% (in bold), and the average measurement error was 1.15% (in bold). The results show that the differences between the AV and EV values of the proposed algorithm were very small. By comparison, the maximum measurement error by CHT was 26.45% (in bold), the minimum error was 0.84% (in bold), and the average measurement error was 8.31% (in bold). The comparison results indicate that the proposed algorithm outperforms CHT in measuring diameters.

3.3.2. Verification of Center Coordinates

To better evaluate the accuracy of the proposed method for calculating the center coordinates of mushrooms, we introduced the two-dimensional coordinate deviation rate (CDR) [5]:

$$CDR = \left(\left| \frac{c_m - c_a}{w} \right| + \left| \frac{r_m - r_a}{h} \right| \right) \times 100\% \quad (8)$$

where c_m and r_m represent the column and row coordinates of the mushroom's center point measured manually; c_a and r_a represent the column and row coordinates of the mushroom's center point determined by algorithm; and $w = 450$ and $h = 600$ are the width and height of the image, respectively. The mushroom images in Figure 2 were utilized to verify the algorithm performance. Three examples of calculating the CDR are shown in Figure 16, and the calculation results of the proposed algorithm and CHT are illustrated in Tables 6 and 7.

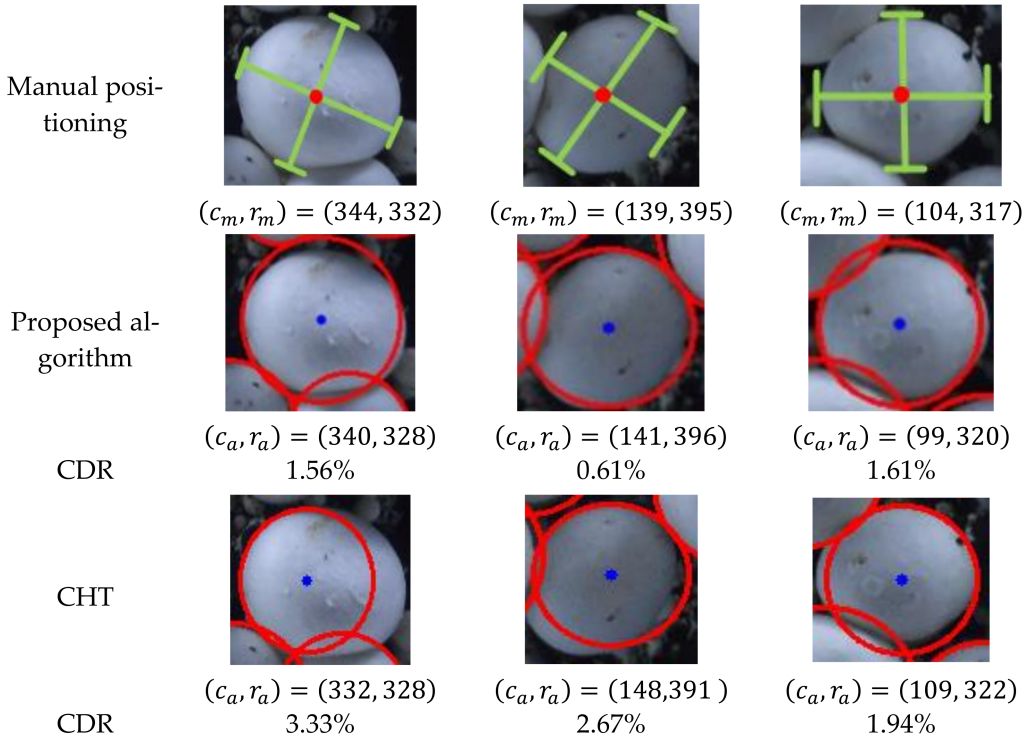


Figure 16. Examples of calculating CDR.

Table 6. CDR results of the proposed algorithm.

NO.	**1**	**2**	**3**	**4**	**5**
Manual positioning	(318, 10)	(295, 184)	(315, 253)	(386, 234)	(344, 332)
Proposed algorithm	(313, 16)	(292, 180)	(313, 252)	(381, 230)	(340, 328)
CDR (%)	2.11	1.33	**0.61**	1.78	1.56
NO.	6	7	8	9	10
Manual positioning	(359, 406)	(331, 478)	(334, 535)	(281, 399)	(220, 344)
Proposed algorithm	(357, 403)	(326, 473)	(331, 534)	(278, 401)	(215, 342)
CDR (%)	0.94	1.94	0.83	1.00	1.44
NO.	11	12	13	14	15
Manual positioning	(139, 395)	(62, 387)	(104, 317)	(49, 260)	(21, 572)
Proposed algorithm	(141, 396)	(56, 385)	(99, 320)	(41, 251)	(22, 569)
CDR (%)	0.61	1.67	1.61	**3.28**	0.72
Average CDR: **1.43%**					

Table 7. CDR results of the CHT.

NO.	1	2	3	4	5
Manual positioning	(318, 10)	(295, 184)	(315, 253)	(386, 234)	(344, 332)
CHT	miss	(282, 181)	miss	(385, 229)	(332, 328)
CDR (%)		**3.39**		1.06	3.33
NO.	6	7	8	9	10
Manual positioning	(359, 406)	(331, 478)	(334, 535)	(281, 399)	(220, 344)
The CHT	(354, 402)	(322, 471)	miss	(288, 403)	(218, 341)
CDR (%)	1.78	3.17		2.22	**0.94**
NO.	11	12	13	14	15
Manual positioning	(139, 395)	(62, 387)	(104, 317)	(49, 260)	(21, 572)
CHT	(148, 391)	(67, 383)	(109, 322)	(45, 255)	miss
CDR (%)	2.67	1.78	1.94	1.72	
Average CDR: **2.18%**					

Table 6 shows that the maximum CDR of the proposed algorithm was 3.28% (in bold), the minimum CDR was 0.61% (in bold), and the average CDR was 1.43% (in bold). Table 7 shows that the maximum CDR of CHT was 3.39 % (in bold), the minimum CDR was 0.94% (in bold), and the average CDR was 2.18% (in bold). If the CDR is less than 3%, it is judged that the coordinates of the mushroom's center point were successfully measured. The results show that the proposed algorithm is more effective than CHT in determining the center of a mushroom. For the proposed algorithm, incorrectly treating the stalk as part of the cap was the leading cause of CDR errors. For the CHT, irregularly circular mushrooms tended to cause significant CDR errors. In Table 6, there is one CDR with a value larger than 3, indicating one mushroom with significant errors in the center coordinates. Similarly, Table 7 illustrates three mushrooms with large errors in the center coordinates. The results are consistent with Figure 12b,c.

From the experimental results, we can summarize that the proposed algorithm has high accuracy in mushroom recognition compared to CHT, and it yields better outcomes in measuring the center coordinates and cap diameters of *A. bisporus*.

4. Conclusions

This study proposes a segmentation and recognition algorithm for *A. bisporus*, which can recognize mature *A. bisporus* and obtain center coordinates and cap diameters. A background filtering method based on morphological operations was developed to filter the complex matrix background and extract the foreground images containing only *A. bisporus*. Moreover, a marker-controlled watershed algorithm was used to segment adherent mushrooms. A preliminary segmentation method based on pixel operation and an iterative watershed marker generation method were proposed, improving the algorithm's accuracy of segmentation and recognition. The experimental results show that the average correct detection rate of the proposed algorithm is 95.7%, the average diameter measurement error is 1.15 %, and the average coordinate deviation rate is 1.43%, showing better outcomes than CHT. Based on this algorithm, vision-based mechanical harvesting equipment for *A. bisporus* can be further studied. Moreover, the methods in this paper can be expanded to other vision-based applications in the mushroom industry, such as grading of mushroom quality and growth-rate measurement. On average, the algorithm takes 705.7 ms to process one image. The algorithm is time-consuming because it requires many calculations. Our future work will improve the algorithm's efficiency and shorten the processing time, thereby enhancing its real-time performance.

Author Contributions: Conceptualization, F.D., C.C. and B.Z.; methodology, F.D., C.C. and S.Y.; software, C.C., S.Y. and B.Z.; validation, S.Y., J.M. and F.W.; investigation, J.M. and F.W.; writing—original draft preparation, C.C. and S.Y.; writing—review and editing, F.D.; visualization, S.Y. and J.M.; supervision, F.D.; project administration, C.C., B.Z. and F.D.; funding acquisition, C.C. and B.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research were funded by the Jiangsu Agricultural Science and Technology Innovation Fund (grant No. CX(21)3150 and No. CX(22)3098) and the Lvyangjinfeng Talent Program of Yangzhou (grant No.YZLYJF2020PHD048).

Data Availability Statement: If scholars need more specific data, they can email the corresponding author or the first author.

Acknowledgments: The authors would like to thank Fudong Li for his insightful guidance and Wei Li for his meaningful discussion.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Reis, F.S.; Barros, L.; Martins, A.; Ferreira, I.C.F.R. Chemical composition and nutritional value of the most widely appreciated cultivated mushrooms: An inter-species comparative study. *Food Chem. Toxicol.* **2012**, *50*, 191–197. [CrossRef] [PubMed]
2. Ramos, M.; Burgos, N.; Barnard, A.; Evans, G.; Preece, J.; Graz, M.; Ruthes, A.C.; Jiménez-Quero, A.; Martínez-Abad, A.; Vilaplana, F.; et al. *Agaricus bisporus* and its by-products as a source of valuable extracts and bioactive compounds. *Food Chem.* **2019**, *292*, 176–187. [CrossRef]
3. Büchner, R.; Vörös, M.; Allaga, H.; Varga, A.; Bartal, A.; Szekeres, A.; Varga, S.; Bajzát, J.; Bakos-Barczi, N.; Misz, A.; et al. Selection and Characterization of a Bacillus Strain for Potential Application in Industrial Production of White Button Mushroom (*Agaricus bisporus*). *Agronomy* **2022**, *12*, 467. [CrossRef]
4. Huang, M.; He, L.; Choi, D.; Pecchia, J.; Li, Y. Picking dynamic analysis for robotic harvesting of *Agaricus bisporus* mushrooms. *Comput. Electron. Agric.* **2021**, *185*, 106145. [CrossRef]
5. Yang, S.; Ni, B.; Du, W.; Yu, T. Research on an Improved Segmentation Recognition Algorithm of Overlapping *Agaricus bisporus*. *Sensors* **2022**, *22*, 3946. [CrossRef]
6. Shi, R.; Dai, F.; Zhao, W.; Li, Z.; Xie, X.; Li, Y. Design and test of self-walking ruffling machine in factory production of *Agaricus bisporus*. *J. Gansu Agric. Univ* **2019**, *54*, 176–181.
7. Yang, J.; Zhao, W.; Li, Z.; Xin, S.; Zhang, F. Design and experiment of key assorted device based on factory production of *Agaricus bisporus*. *Trans. Chin. Soc. Agric. Eng.* **2018**, *34*, 43–51.
8. Zhu, X.; Zhao, K.; Ji, J.; Sun, J.; Ma, H. Design and experiment of environmental factor control system for industrialized production of *Agaricus bisporus*. *J. Agric. Mech. Res.* **2021**, *43*, 156–162.
9. Zied, D.C.; Pardo-Giménez, A. (Eds.) *Edible and Medicinal Mushrooms: Technology and Applications*; John Wiley & Sons: Chichester, UK, 2017.
10. Van Griensven, L.J.L.D. The Cultivation of Mushrooms: Its Present Status and Future Developments. *Outlook Agric.* **1987**, *16*, 131–135. [CrossRef]
11. Tillett, R.D.; Batchelor, B.G. An algorithm for locating mushrooms in a growing bed. *Comput. Electron. Agric.* **1991**, *6*, 191–200. [CrossRef]
12. Yu, G.; Luo, J.; Zhao, Y. Region marking technique based on sequential scan and segmentation method of mushroom images. *Trans. Chin. Soc. Agric. Eng.* **2006**, *22*, 139–142. [CrossRef]
13. Qiu, J. Object detection algorithm for the mushroom picking robot. *J. Inner Mongolia Agric. Univ.* **2021**, *42*, 89–92.
14. Ji, J.; Sun, J.; Jin, X.; Ma, H.; Zhu, X. Measuring the Cap Diameter of White Button Mushrooms (*Agaricus bisporus*) by Using Depth Image Processing. *Appl. Eng. Agric.* **2021**, *37*, 623–633. [CrossRef]
15. Masoudian, A. Computer Vision Algorithms for an Automated Harvester. Master's Thesis, University of Western Ontario, London, ON, Canada, 2013.
16. Lee, C.-H.; Choi, D.; Pecchia, J.; He, L.; Heinemann, P. Development of A Mushroom Harvesting Assistance System using Computer Vision. In Proceedings of the 2019 ASABE Annual International Meeting, St. Joseph, MI, USA, 7–10 July 2019.
17. Lu, C.-P.; Liaw, J.-J.; Wu, T.-C.; Hung, T.-F. Development of a Mushroom Growth Measurement System Applying Deep Learning for Image Recognition. *Agronomy* **2019**, *9*, 32. [CrossRef]
18. Lu, C.-P.; Liaw, J.-J. A novel image measurement algorithm for common mushroom caps based on convolutional neural network. *Comput. Electron. Agric.* **2020**, *171*, 105336. [CrossRef]
19. Cao, J. RESEARCH on Computer Vision System of *Agaricus bisporus* Harvesting Robot Based on Deep Learning. Ph.D. Dissertation, Chinese Academy of Agricultural Sciences, Beijing, China, 2021.
20. Saddik, A.; Latif, R.; El Ouardi, A. Low-Power FPGA Architecture Based Monitoring Applications in Precision Agriculture. *J. Low Power Electron. Appl.* **2021**, *11*, 39. [CrossRef]

21. Saddik, A.; Latif, R.; El Ouardi, A.; Alghamdi, M.I.; Elhoseny, M. Improving Sustainable Vegetation Indices Processing on Low-Cost Architectures. *Sustainability* **2022**, *14*, 2521. [CrossRef]
22. Saddik, A.; Latif, R.; Taher, F.; El Ouardi, A.; Elhoseny, M. Mapping Agricultural Soil in Greenhouse Using an Autonomous Low-Cost Robot and Precise Monitoring. *Sustainability* **2022**, *14*, 15539. [CrossRef]
23. Saddik, A.; Latif, R.; Elhoseny, M.; El Ouardi, A. Real-time evaluation of different indexes in precision agriculture using a heterogeneous embedded system. *Sustain. Comput. Inform. Syst.* **2021**, *30*, 100506. [CrossRef]
24. Pizer, S.M.; Amburn, E.P.; Austin, J.D.; Cromartie, R.; Geselowitz, A.; Greer, T.; ter Haar Romeny, B.; Zimmerman, J.B.; Zuiderveld, K. Adaptive histogram equalization and its variations. *Comput. Vis. Graph. Image Process.* **1987**, *39*, 355–368. [CrossRef]
25. Nithyananda, C.R.; Ramachandra, A.C.; Preethi. Review on Histogram Equalization based Image Enhancement Techniques. In Proceedings of the 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, India, 3–5 March 2016; pp. 2512–2517.
26. Xu, X.; Xu, S.; Jin, L.; Song, E. Characteristic analysis of Otsu threshold and its applications. *Pattern Recognit. Lett.* **2011**, *32*, 956–961. [CrossRef]
27. Serra, J.; Soille, P. *Mathematical Morphology and Its Applications to Image Processing*; Springer: Dordrecht, The Netherlands, 1994.
28. Pal, S.; Chatterjee, S.; Dey, D.; Munshi, S. Morphological operations with iterative rotation of structuring elements for segmentation of retinal vessel structures. *Multidimens. Syst. Signal Process.* **2018**, *30*, 373–389. [CrossRef]
29. Li, H.; Asbjörnsson, G.; Lindqvist, M. Image Process of Rock Size Distribution Using DexiNed-Based Neural Network. *Minerals* **2021**, *11*, 736. [CrossRef]
30. Huang, H.-W.; Li, Q.-T.; Zhang, D.-M. Deep learning based image recognition for crack and leakage defects of metro shield tunnel. *Tunn. Undergr. Space Technol.* **2018**, *77*, 166–176. [CrossRef]
31. Bleau, A.; Leon, L. Watershed-Based Segmentation and Region Merging. *Comput. Vis. Image Underst.* **2000**, *77*, 317–370. [CrossRef]
32. Vincent, L.; Soille, P. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Trans. Pattern Anal. Mach. Intell.* **1991**, *13*, 583–598. [CrossRef]
33. Haris, K.; Efstratiadis, S.; Maglaveras, N.; Katsaggelos, A. Hybrid image segmentation using watersheds and fast region merging. *IEEE Trans. Image Process.* **1998**, *7*, 1684–1699. [CrossRef]
34. Wu, W.; Zhou, L.; Chen, J.; Qiu, Z.; He, Y. GainTKW: A Measurement System of Thousand Kernel Weight Based on the Android Platform. *Agronomy* **2018**, *8*, 178. [CrossRef]
35. Rao, K.N.R.M.; Dempster, A.G. Modification on distance transform to avoid over-segmentation and under-segmentation. In Proceedings of the International Symposium on VIPromCom Video/Image Processing and Multimedia Communications, Zadar, Croatia, 16–19 June 2002; pp. 295–301.
36. Shih, F.-C.; Mitchell, O. A mathematical morphology approach to Euclidean distance transformation. *IEEE Trans. Image Process.* **1992**, *1*, 197–204. [CrossRef]
37. Sabzi, S.; Pourdarbani, R.; Rohban, M.; García-Mateos, G.; Paliwal, J.; Molina-Martínez, J. Early Detection of Excess Nitrogen Consumption in Cucumber Plants Using Hyperspectral Imaging Based on Hybrid Neural Networks and the Imperialist Competitive Algorithm. *Agronomy* **2021**, *11*, 575. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.