2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Strawberry Weight Estimation Based on Plane-Constrained Binary Division Point Cloud Completion

Yanjiang Huang, Jiepeng Liu and Xianmin Zhang

Abstract— **Labor shortages and the development of digital technology both impose requirements on the fruit industry. Modern agricultural competition has shifted from competition between products to competition between supply chains. Enhancing the digitization of production lines is crucial for gaining a competitive advantage. Strawberries, as fruits with a short shelf life, require sorting and packaging of fruits of different weights after being harvested. Estimating strawberry weight through visual technology can save time and labor costs. Common methods include methods based on feature size and learning-based methods, with the former having larger errors and the latter requiring a large amount of data. To address these issues, we propose a dataset for estimating strawberry weight, which includes strawberries with different heights and angles. Additionally, we propose a strawberry weight estimation method based on plane-constrained binary division point cloud completion. This method separates the plane point cloud and strawberry point cloud, constructs a coordinate system on the strawberry point cloud, generates an axis-aligned bounding box (AABB), and estimates the strawberry weight based on the bounding box and placement plane as constraints. Through comparison with different methods, we achieved a maximum improvement of 20.95% in prediction accuracy, demonstrating that our method provides the best estimation accuracy.**

I. Introduction

The aging population has led to a long-term shortage of labor, and the occurrence of unexpected events such as epidemics can result in a short-term shortage of labor. These factors have a significant impact on agricultural production, potentially leading to a decline in the quality of agricultural products and price increases. Technological advancements have brought about the ability to mitigate risks in agricultural production by replacing various processes with robots and automated equipment, improving the quality and yield of crops, and freeing laborers from the heavy burden of agricultural production.

Machine vision, which simulates and reproduces intelligent human visual behaviors through computers, processes and understands information extracted from images of objective objects, and is ultimately applied in practical detection and control. In many cases in agricultural production, information about plant growth and fruit ripeness needs to be obtained through human eyes, making machine vision increasingly applicable in agriculture.

Strawberries, as a high-value crop, have seen a continuous increase in global production, reaching 8.885 million tons in 2022 [1]. As a fruit with a short shelf life, it is crucial for strawberries to undergo as short a period as possible from harvesting to sale to ensure freshness for consumers. There has been a lot of research focused on strawberries, including gripper design for strawberry-picking robots [2], strawberry fruit recognition [3]-[5], path planning during harvesting [6], and weight estimation [23]. Strawberries are generally sold and classified according to their weight. Traditional methods of manual weighing are costly, while weight estimation methods based on vision technology lower labor costs. However, accurate weight estimation is required to ensure the interests of both consumers and sellers.

To address these issues, this paper proposes a strawberry weight estimation method based on plane-constrained binary division point cloud completion, which includes point cloud preprocessing and coordinate system construction, binary division to complete point cloud and estimate weight. The weight is accurately estimated by completing the strawberry point cloud. The main contributions of this paper are as follows: (1) We propose a new dataset for strawberry weight estimation, which includes RGB images, depth images, and point clouds of strawberries at different heights and angles, facilitating comparisons among different methods for estimating strawberry weight. (2) We propose a strawberry weight estimation method based on the plane-constrained binary division point cloud completion, which accurately estimates weight by reconstructing the complete strawberry point cloud from local strawberry point clouds and calculating the convex hull weight. (3) Through experimental comparisons with the state-of-the-art (SOTA) methods, we demonstrate that our method achieves higher estimation accuracy.

II. Related Works

The estimation of weight or volume based on machine vision has always been a hot research topic. It can be divided into traditional methods based on feature size fitting and learning-based methods, each with its own advantages and disadvantages. If the fruit has a relatively regular shape, the feature size fitting method can achieve good results and is easy to implement. However, its accuracy is poor for estimating complex shapes. The estimation effectiveness of learning-based methods depends on the adaptability of the model and the quality of the dataset.

A. Methods Based on Feature Size Fitting

The method based on feature size fitting obtains the feature size of the fruit through image processing. It can directly fit the weight or volume prediction function or estimate the

This work was supported in part by the National Natural Science Foundation of China under Grant 52075178 and Grant 52130508; and the Opening Project of Key Laboratory of Safety of Intelligent Robots for State Market Regulation, PR China under Grant GQI-KFKT202201.
(Corresponding author: Xianmin Zhang.)
Yanjiang Huang, Jiepeng Liu, and Xianmin Zhang are with the Guangdong Provincial Key Laboratory of Precision Equipment and Manufacturing Technology, South China University of Technology, Guangzhou 510640, China (e-mail: mehuangyj@scut.edu.cn; 202121002124@mail.scut.edu.cn; zhangxm@scut.edu.cn).

volume directly using area, thus estimating the weight. A method using ellipsoidal approximation and image processing was used to estimate the volume of watermelons of different sizes. The disk approximation method was used to estimate the volume of each watermelon surface image captured by a low-cost CMOS camera, providing a simple, fast, and non-invasive method for watermelon volume estimation [7]. The RGB-depth (RGB-D) sensor was used to measure the maximum diameter and volume of sweet onions. The density of the onions was estimated using the measured parameters, demonstrating that the depth image estimation of onion diameter has higher average accuracy and robustness than color images [8]. An automatic apple sorting machine was designed and implemented, which captures at least four images for each apple. The area is calculated based on the images, and a second-degree polynomial estimation of apple weight using only the apple's area is studied [9]. Similarly, for the weight estimation problem of apples, a model established using Canny edge detection method can identify the weight of the apples. The weight ratio is calculated by comparing the actual weight on the physical object with the weight on the image [10]. Star fruits were captured using a camera, then processed using machine vision and image processing techniques. The star fruits were sliced numerically along the vertical axis, and the physical properties of each slice were estimated using the disk and cone methods. The total volume was calculated by accumulating the volumes of each slice, multiplied by the density coefficient to obtain the weight [11]. This process is similar in principle to [7].

There is a lot of research on estimating the weight or volume of mangoes. The study showed that three most distinctive dimensions, namely, photometric length, maximum width, and maximum thickness, are sufficient to describe the size and weight characteristics of mangoes [12]. However, due to significant variations in size and shape among different mango varieties, the correlations found in [12] cannot be applied to other mango varieties. The volume of mangoes was estimated using the spatial geometric analysis method. The total volume is the sum of the volumes of all the sections formed by cross-cutting the fruit along its length. If the section intervals are small enough, the volume of each section can be approximated as an elliptical cylinder volume by binarizing the mango images and calculating the minor axis length of each section [13]. A region-based global threshold color binarization method, combined with median filtering and morphological analysis, was proposed to divide mangoes into three mass grades: large, medium, and small based on projection area and Feret diameter [14]. The visible surface area of the mango was determined using curve fitting and the area between two curves method. The required thickness value was calculated based on the correlation equation between width and thickness. The thickness variation along the length of the mango was modeled as an elliptical function. The overall estimated volume of the mango was then calculated using the volume of the elliptical cylinder from the cross-sections and the length [15].

In addition to the estimation of weight or volume of fruits, the weight estimation of animals also has important research value. Similar to [8] and [12], the weight of fish was calculated by extracting dimensions such as length, width, and height through image processing [16].

B. Learning-based Methods

With the development of machine learning, deep learning, and other technologies, learning-based methods for weight or volume estimation can also provide excellent estimation results.

By drawing lines on strawberry images, the shape features of the strawberries are obtained, and K-means clustering is used to classify the weight of strawberry images [17]. Fruit images are subjected to binarization to separate the fruit images from the background. The number of pixels covering the fruit images is calculated to estimate their size, and a fuzzy rule-based algorithm is used to grade the fruit quality [18]. Support Vector Machines (SVM) have shown many unique advantages in solving small-sample, nonlinear, and high-dimensional pattern recognition, making them widely used in fruit grading applications [19]-[22]. Depth images of tomatoes in different directions are obtained using image processing techniques, features are extracted, and five regression prediction models based on 2D and 3D image features are established. Experimental results demonstrate that the Radial Basis Function Support Vector Machine (RBF-SVM) model is more accurate than other comparison models [19]. Seven geometric attributes of sweet lime are extracted using image processing, and a Support Vector Machine Regression (SVMR) modeling technique is used to establish a weight estimation model for fruit samples [20]. Image processing methods are used to extract traditional 2D image features (projection area) and 3D height information. These two types of combined features are used to train the Least Squares Support Vector Machine (LS-SVM) model as volume and weight prediction models [21]. The application of discriminant analysis and SVM in mango shape classification is studied [22]. Multiple features of strawberries are extracted to train a random forest model with decision trees, and this model is used to predict the weight of strawberries [23].

Neural networks used for object detection have also been applied to fruit volume or weight estimation. The You Only Look Once (YOLOv5) neural network algorithm is used to classify fruits after detection [24]. A deep learning regression model combining U-Net, Visual Geometry Group Network (VGG-16), and attention modules is proposed, which can predict the weight of grape clusters in images with small errors [25]. Similarly, neural networks have also been applied to animal weight estimation. A neural network-based method for pig weight detection and measurement is proposed. Eight methods including color, texture, centroid, major axis, minor axis, eccentricity, and area are used to extract pig features, and the neural network is used for recognition based on statistical data in the original database [26].

III. DATASET

The dataset collection system includes a RealSense SR300 depth camera, Robotiq 2f-85 gripper, UR5 robotic arm, and a placement plane. The gripper is installed on the end of the UR5 robotic arm to install the depth camera (see Fig. 1). The optimal working distance for depth cameras is 0.2~0.5m. The robotic arm is used to move the camera to capture RGB images, depth images, and point clouds of strawberry placed on the plane. The point clouds serve as the input for our method, while the RGB images and depth images are used as input for the comparative method.

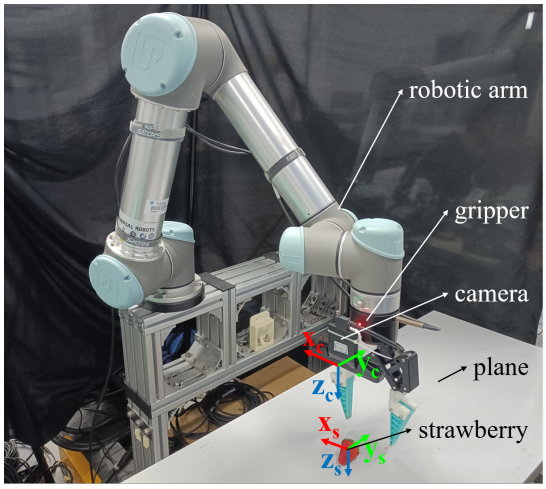

Fig. 1. The dataset collection system.

The dataset collection process is as follows: Firstly, the depth camera captures the point cloud and performs plane segmentation to obtain the normal vector of the plane. The robotic arm is adjusted to make the z-axis of the depth camera coordinate system perpendicular to the plane and at a distance of 0.25m from the plane. Secondly, a single strawberry is placed directly below the camera, and the current rotation angle of the end joint of the robotic arm is defined as 0 degrees. The end joint is then rotated from 0 to 90 degrees in increments of 15 degrees at 7 different positions to collect RGB images, depth images, and point cloud data. The end joint is then returned to 0 degrees, and the robotic arm moves to two positions at 0.35m and 0.45m with the end joint rotated from 0 to 90 degrees to collect data. Finally, the collected RGB images, depth images, point clouds, and strawberry weight from the 21 strawberries are used as the data for the current strawberry, and the process is repeated for the next strawberry.

We have prepared 20 strawberries with weights ranging from 10 to 20 grams, and collected 21 data samples for each strawberry at different heights and rotation angles. This amounts to a total of 420 data samples to construct the strawberry weight dataset. This dataset serves as the fitting data for our proposed method and the training data for the comparative method.

IV. Proposed Method

Currently, picked strawberries are placed in small baskets and then manually sorted into different baskets based on their weight for sale. However, the manual sorting process has a large margin of error. Strawberry weight estimation based on visual provides a non-contact method to accurately estimate the weight of strawberries. Currently, there are two known methods for visual based strawberry weight estimation. One method fits an approximate function based on the strawberry's characteristic size and weight, but this method has a large error. The other method is a learning-based weight estimation method, which provides higher accuracy but requires a high-quality dataset. In this paper, we propose a strawberry weight estimation method based on plane-constrained binary division point cloud completion. The complete framework of the method is shown in Fig. 2. By reconstructing the local point cloud, we obtain the actual volume of the strawberry. Since the density of the strawberry remains roughly constant, we can estimate the weight by multiplying the density by the volume. Our proposed method consists of two main parts. The first part involves point cloud preprocessing and coordinate system construction, which prepares for point cloud completion. The second part is the binary division point cloud completion, which estimates the weight of the strawberry by completing the point cloud.

A. Point Cloud Preprocessing and Coordinate System Construction

Since the input point cloud contains strawberry and a placement plane, a plane segmentation algorithm based on RANSAC (Random Sample Consensus) is used to separate the strawberry point cloud from the plane point cloud. The RANSAC algorithm randomly selects a subset of samples from the point cloud, uses the least squares method to calculate the plane model of $Ax + By + Cz + D = 0$, and then calculates the deviation of all points from the model. The deviation is compared with a set threshold, and when the deviation is less than the threshold, the sample point belongs to the inlier of the model; otherwise, it is considered as an outlier. The algorithm records the model parameters with the maximum number of inliers, as well as the number of inliers, and then repeats this process until the desired error rate, number of inliers, or the current iteration number meets the requirements. The result is the strawberry point cloud, plane point cloud, and plane model parameters.

The obtained strawberry point cloud contains noisy points, which have a significant impact on subsequent point cloud completion. Therefore, outlier filtering is used to remove the noisy points from the strawberry point cloud. The outlier filtering algorithm looks for all neighboring points of each point; calculates the distance d_{ij} between each point and its neighbors, where $i = [1, \ldots, n]$ represents the total of n points and $j = [1, \ldots, k]$ represents k neighbors of each point. The distance parameters are modeled using a Gaussian distribution $X \sim N(\mu, \sigma)$, and μ (the mean of distance) and σ (the standard deviation of distance) of all points with their neighbors are calculated as follows:

$$\mu = \frac{1}{nk}\sum_{i=1}^{n}\sum_{j=1}^{k} d_{ij} \tag{1}$$

$$\sigma = \sqrt{\frac{1}{nk}\sum_{i=1}^{n}\sum_{j=1}^{k}(d_{ij} - \mu)^2} \tag{2}$$

Calculate the average distance from each point to its neighbors as follows:

$$\bar{d}_i = \frac{1}{k}\sum_{j=1}^{k} d_{ij} \tag{3}$$

Iterate through all points, if the average distance is greater than the specified confidence level of the Gaussian distribution, remove it, for example: $\bar{d}_i > \mu + 3\sigma$ or $\bar{d}_i < \mu - 3\sigma$.

The reference coordinate system of the original point cloud is the camera coordinate system. In order to facilitate subsequent operations, a strawberry coordinate system is fixed to the filtered strawberry point cloud. The origin of the strawberry coordinate system is calculated as follows:

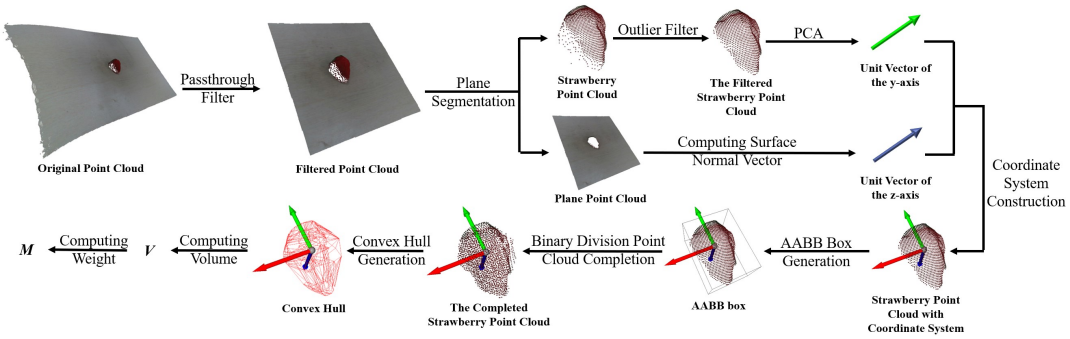

Fig. 2. Framework for weight estimation method based on plane-constrained binary division point cloud completion.

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{4}$$

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{5}$$

$$\bar{z} = \frac{1}{n}\sum_{i=1}^{n} z_i \tag{6}$$

Where (x_i, y_i, z_i) is the coordinate of the i-th point in the camera coordinate system, and the coordinate of the origin of the strawberry coordinate system in the camera coordinate system is $(\bar{x}, \bar{y}, \bar{z})$.

To determine a right-hand coordinate system, it is necessary to know the unit vectors of two axes. The unit vector v_z of the z-axis in the strawberry coordinate system is the normal vector of the plane, namely $(\frac{A}{\sqrt{A^2+B^2+C^2}}, \frac{B}{\sqrt{A^2+B^2+C^2}}, \frac{C}{\sqrt{A^2+B^2+C^2}})$, with its positive direction pointing towards the plane; Due to the long conical shape of strawberry, principal component analysis (PCA) can be used to obtain the main direction of point distribution, which serves as the y-axis of the strawberry coordinate system. The PCA calculation process involves constructing a covariance matrix:

$$S = \frac{1}{n-1}\sum_{i-1}^{n}(p_i - \bar{p})(p_i - \bar{p})^T \tag{7}$$

Where p_i is the coordinate of the i-th point, $\bar{p}$ is the center of the strawberry point cloud. Perform eigenvalue decomposition on S, assuming that the eigenvalues of S are $\lambda_1 > \lambda_2 > \lambda_3$. The corresponding feature vectors v_1, v_2, v_3, and v_m represent the projection vector of point in the m-th principal direction, and v_1 is used as the y-axis unit vector v_y of the strawberry coordinate system. Since the x-axis unit vector of the strawberry coordinate system is calculated as follows:

$$v_x = v_y \times v_z \tag{8}$$

However, this coordinate system is not unique because its y-axis direction can be from the strawberry stem to the strawberry tip or from the strawberry tip to the strawberry stem. Calculate the y-axis position of the strawberry point cloud with the largest x-axis span in this coordinate system. If it is in the positive y-axis direction, the coordinate system does not need to be changed. Otherwise, rotate the strawberry coordinate system 180° around its z-axis to obtain the final coordinate system. The coordinate systems mentioned later in this article are all strawberry coordinate systems.

Using the strawberry coordinate system as the reference coordinate system, generate the AABB box of the strawberry point cloud. Each side of the AABB box is parallel to a plane in the strawberry coordinate system. This is achieved by calculating the maximum values x_{max}, y_{max}, z_{max} and minimum values x_{min}, y_{min}, z_{min} of the strawberry point cloud along the three axes of the coordinate system.

B. Binary Division to Complete Point Cloud and Estimate Weight.

Strawberry have a nearly symmetrical shape and can be divided into roughly equal halves by a plane. By using a symmetrical plane, the local point cloud of the strawberry can be reconstructed into a complete point cloud. Since the strawberry is placed on a plane, influenced by gravity, it contacts with the plane. This allows the plane of placement to serve as a constraint for constructing the symmetrical plane. To construct a symmetrical plane, three points need to be determined. The first point is the one with the minimum value on the y-axis of the strawberry point cloud. The y-coordinates of the other two points are y_{max}, and their x-coordinates are x_{max} and x_{min}, respectively. The z-coordinate is calculated using the binary division method, starting with the upper boundary $z_{upper} = z_{plane}$, where z_{plane} is the distance from the placement plane to the origin of the strawberry coordinate system. The initial lower boundary of the z-coordinate is $z_{lower} = z_{min}$. Then, the middle value is calculated using the binary search method:

$$z_{middle} = \frac{z_{upper} + z_{lower}}{2} \tag{9}$$

So, the coordinates of the other two points on the symmetric plane are respectively $(x_{min}, y_{max}, z_{middle})$ and $(x_{max}, y_{max}, z_{middle})$. The initial symmetric plane is constructed using these three points, and the strawberry point cloud is mirrored with reference to this plane to obtain the mirrored strawberry point cloud. The distance between the point with the maximum z-axis value in the mirrored strawberry point cloud and the placement plane is then calculated to check if it is within the set threshold. If it is, the iteration stops and the mirrored strawberry point cloud is merged with the original strawberry point cloud to form the completed strawberry point cloud. Otherwise, it is determined whether to change the upper or lower boundary based on whether the z-value of the mirrored strawberry point cloud is greater than the z-value of the

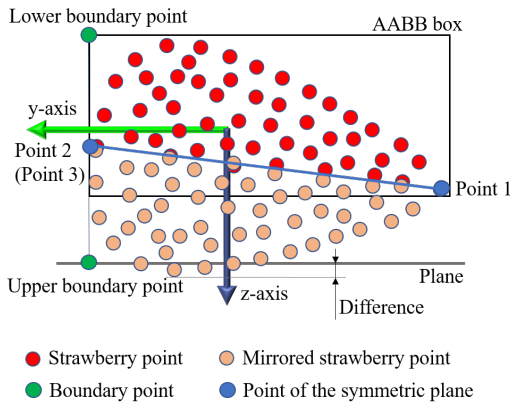

Fig. 3. Plane-constrained binary division point cloud completion. The perspective in this figure is looking in the opposite direction of the x-axis in the strawberry coordinate system. The blue point, Point 1, represents the first point of the symmetrical plane. The other two points, Point 2 and Point 3, are determined by taking the midpoint of the z-values of the lower boundary point and upper boundary point, which are represented by green point on the figure. Point 2 and Point 3 overlap in this perspective. The symmetrical plane is then constructed using these three points, and the red strawberry point cloud is mirrored to generate light pink mirror points. Whether to continue the iteration is determined by the distance between the mirror point with the maximum z-value and the plane.

placement plane. If it is greater than, then $z_{upper} = z_{middle}$, otherwise $z_{lower} = z_{middle}$, and then proceed to the next iteration until the difference is less than the threshold to obtain the completed strawberry point cloud (see Fig. 3).

For the completed strawberry point cloud, we assume that it has a similar shape to the corresponding real strawberry, so its volume can be estimated to determine its weight. However, the volume of a point cloud cannot be directly calculated. Therefore, in this study, we utilize a fast convex hull algorithm [27] to generate a convex hull composed of a triangle mesh for the completed strawberry point cloud. In comparison to the Alpha shape algorithm [28] and Ball pivoting algorithm [29], the fast convex hull algorithm is capable of efficiently and effectively constructing a high-quality convex hull. The origin of the strawberry coordinate system is enclosed by the convex hull. The volume V of the convex hull can be obtained by accumulating the volumes of tetrahedrons formed by three points of the triangle mesh and the origin. The calculation formula is as follows:

$$V = \frac{1}{6}\sum_{i=1}^{m}((\vec{p}_{i3} - \vec{p}_{i1}) \times (\vec{p}_{i2} - \vec{p}_{i1}) \bullet \vec{p}_{i1}) \tag{10}$$

$\vec{p}_{i3}, \vec{p}_{i2}, \vec{p}_{i1}$ are the vectors from the origin of the strawberry coordinate system to the three points of the i-th triangle. The weight of the strawberry is proportional to its volume and can be obtained by multiplying the volume V by the density coefficient k.

$$M = kV \tag{11}$$

The density coefficient can be obtained by fitting the actual weight of strawberries to the volume of their convex hull using the least squares method. The above method is called our method 1.

The depth value of strawberry point cloud may also affect the estimated weight value, so it is added to the weight fitting,

$$M = k_1 V + k_2 D \tag{12}$$

Where D is the z-value at the center of the strawberry point cloud. The above method is called our method 2.

V. Results and Analysis

The algorithm proposed in this article has been implemented in Python 3.6. Point cloud operations are achieved using Open3d 0.9.0 and NumPy 1.19. The algorithm was run on Ubuntu 16.04 with an Intel(R) Core(TM) i5-9400F CPU @ 2.90GHz. The algorithm is not utilize GPU.

To measure the accuracy of weight estimation, we use the Protocol for Percentage of Correct Weight (PCW) proposed by [23]. This standard measures the regression error as a percentage relative to the actual weight. If the error falls within the expected tolerance level, the inference is marked as accurate; otherwise, it is marked as false. The accuracy of the model is evaluated using the predicted percentage within the tolerance level. Similar to [23], in our experiment, the tolerance levels are set at 0.05, 0.10, 0.15, 0.20, 0.25, and 0.30, corresponding to accuracy ranges of 70-95%.

The weight of strawberries is correlated with their maximum diameter and the length from the stem to the tip of the strawberry. Predicting weight based on these feature sizes is a simple strategy. In our experiment, to compare with the method of directly fitting strawberry feature size to weight, we design two methods based on the AABB box generated by the proposed method. One method (XXY method) is to fit the $(x_{max}\text{-}x_{min})^2 \times (y_{max}\text{-}y_{min})$ value of the AABB box in the strawberry coordinate system with the strawberry weight to obtain a prediction function. The other method (XYZ method) is to fit the $(x_{max}\text{-}x_{min}) \times (y_{max}\text{-}y_{min}) \times (z_{plane}\text{-}z_{min})$ value with the strawberry weight to obtain a prediction function. These two prediction functions are used to estimate the weight of strawberries. The latter may have higher accuracy for flat-shaped strawberries compared to the former.

In addition to comparing with the conventional method of fitting feature size, we also compare with the method proposed by [23]. This method first uses the Detectron-2 model to obtain segmented images of individual strawberries to create a segmented point cloud. Then, the point cloud's convex hull is reconstructed, and principal component analysis is performed on the convex hull to determine the major axis. A feature vector composed of strawberry bounding box area, segmentation mask, depth value histogram, principal components, and the actual weight of strawberries is used to train a random forest model with a decision tree. This model is then used to predict the weight of strawberry.

The dataset is divided into two groups in a 1:1 ratio, with one group used for fitting the prediction functions and training the comparative methods, and the other group used for testing and comparing the results. Fig. 4, Fig. 5, Fig. 6, and Fig. 7 respectively show the fitting results of the XXY method, XYZ method, our method 1 and our method 2. From the figure, it can be seen that the point dispersion estimated by the XXY method and XYZ method is greater than ours, and the comparison results can be foreseen. Fig. 8 shows the

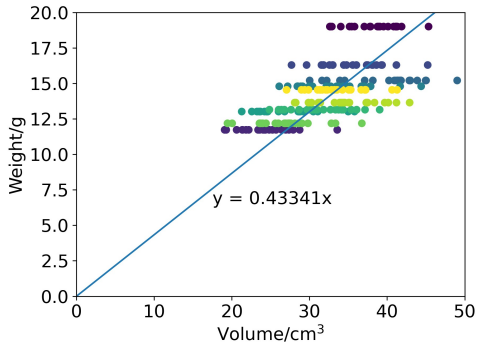

Fig. 4. Fitting results of the XXY method.

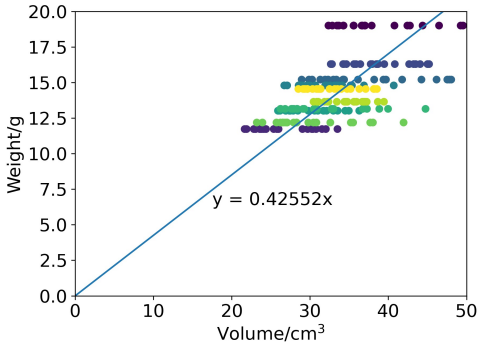

Fig. 5. Fitting results of the XYZ method.

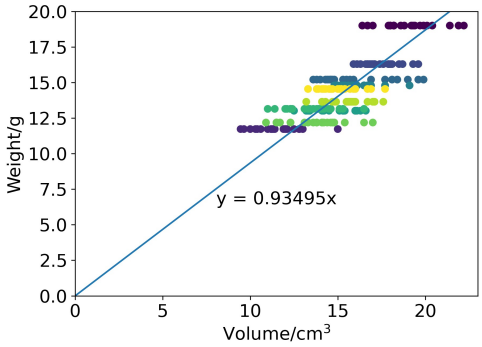

Fig. 6. Fitting results of our method 1.

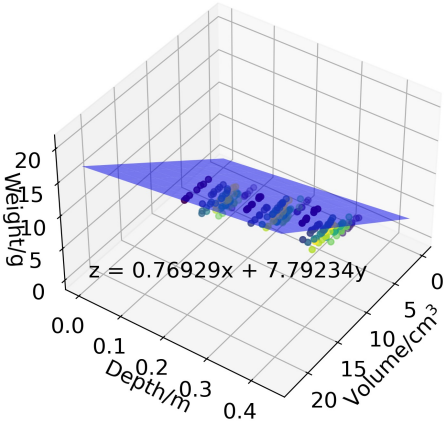

Fig. 7. Fitting results of our method 2.

results of the strawberry weight estimation experiment. Our method 2 achieves the best prediction performance at all

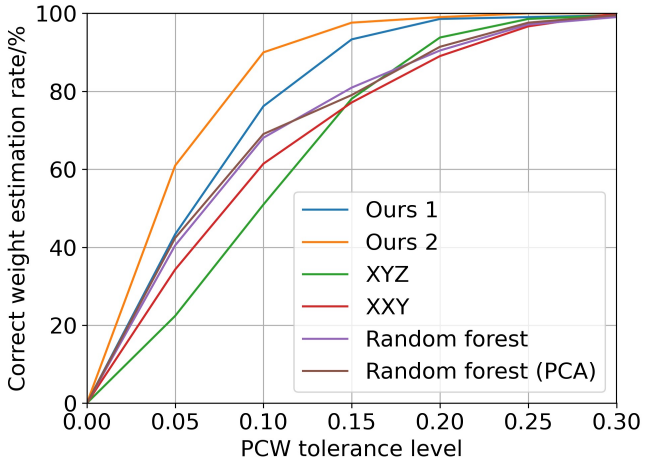

Fig. 8. The results of the strawberry weight estimation experiment.

PCW tolerance levels, with a significant improvement compared to our method 1, indicating that the sampling depth of strawberry point cloud has a significant impact on weight estimation. It achieves an accuracy of 97.62% at PCW@0.15, and it outperforms other models except for our method 1 by at least 18.57%, 20.95%, 16.67%, and 5.24% at PCW@0.05, PCW@0.10, PCW@0.15, and PCW@0.20, respectively. The random forest prediction method also produces good results, surpassing the method of fitting feature size overall. However, its prediction accuracy is lower than the XYZ method at PCW@0.20, possibly due to overfitting. The random forest prediction with PCA shows a slight improvement compared to the one without PCA, but there is a decrease at PCW@0.15, indicating that PCA has a moderate effect on modeling the relationship between strawberry orientation-induced weight changes. The method of simple fitting based on feature size also provides good prediction results, but its accuracy is lower at PCW@0.05 and PCW@0.10 due to the simplicity of the model. The XYZ method is also not superior to the XXY method at each PCW, possibly because when strawberries are placed on a plane, the approximate symmetry axis of the strawberries does not align parallel to the plane, causing the z_{plane}-z_{min} value to not accurately reflect the size of strawberries.

VI. Conclusion

This article presents a new dataset containing strawberry RGB images, depth images, point clouds, and weight, with varying heights and angles. The dataset is designed for research on strawberry weight estimation based on visual or point cloud analysis. We also propose a strawberry weight estimation method that utilizes a plane-constrained binary division algorithm for point cloud completion. By completing the local point clouds of strawberries, we can estimate their complete weights. Our experimental results demonstrate that our method achieves the highest prediction accuracy compared to simple dimension fitting methods or machine learning approaches.

REFERENCES

[1] "World Strawberry Production by Country," *https://www.atlasbig.com/en-us/countries-strawberry-production*, accessed: 2023-Feb-21.

[2] Y. Xiong, P. J. From and V. Isler, "Design and Evaluation of a Novel Cable-Driven Gripper with Perception Capabilities for Strawberry Picking Robots," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia, 2018, pp. 7384-7391.

[3] X. Li, J. Li and J. Tang, "A deep learning method for recognizing elevated mature strawberries," *2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, Nanjing, China, 2018, pp. 1072-1077.

[4] Y. Ge, Y. Xiong, G. L. Tenorio and P. J. From, "Fruit Localization and Environment Perception for Strawberry Harvesting Robots," in *IEEE Access*, vol. 7, pp. 147642-147652, 2019.

[5] Y. Yu, K. Zhang, H. Liu, L. Yang and D. Zhang, "Real-Time Visual Localization of the Picking Points for a Ridge-Planting Strawberry Harvesting Robot," in *IEEE Access*, vol. 8, pp. 116556-116568, 2020.

[6] Y. Xiong, Y. Ge and P. J. From, "Push and Drag: An Active Obstacle Separation Method for Fruit Harvesting Robots," *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, 2020, pp. 4957-4962.

[7] A. B. Koc, "Determination of watermelon volume using ellipsoid approximation and image processing," *Postharvest Biology and Technology*, vol. 2007, no. 45, pp. 366–371, 2007.

[8] W. Wang and C. Li, "Size estimation of sweet onions using consumer-grade RGB-depth sensor," *Journal of Food Engineering*, vol. 142, pp. 153–162, 2014.

[9] M. M. Sofu, O. Er, M. C. Kayacan, B. Cetisli, "Design of an Automatic Apple Sorting System using Machine Vision," *Computers and Electronics in Agriculture*, vol. 127, pp. 395–405, 2016.

[10] A. F. N. Masruriyah, M. H. Ijlal, Rahmat, H. H. Handayani, D. Wahiddin and A. Fauzi, "The Estimating of Nutrient Value in Apples Based on Size Employing the Canny Edge Detection Algorithm," *2022 Seventh International Conference on Informatics and Computing (ICIC)*, Denpasar, Bali, Indonesia, 2022, pp. 1-6.

[11] H. M. Tran, K. T. Pham, T. M. Vo, T. -H. Le, T. T. M. Huynh and S. V. T. Dao, "A New Approach for Estimation of Physical Properties of Irregular Shape Fruit," in *IEEE Access*, vol. 11, pp. 46550-46560, 2023.

[12] W. Spreer, J. Muller, "Estimating the Mass of Mango Fruit (Mangifera indica, cv. Chok Anan) from its Geometric Dimensions by Optical Measurement," *Computers and Electronics in Agriculture*, vol. 75, No. 1, pp. 125–131, 2011.

[13] A. M. Bermúdez, G. S. Torres, D. B. Padilla, "Image Analysis for Automatic Feature Estimation of the Mangifera Indica Fruit," *Ingenieriay Desarrollo*, vol. 31, No. 1, pp. 84–104, 2013.

[14] M. A. Momin, M. T. Rahman, M. S. Sultana, C. Igathinathane, A. T. M. Ziauddin and T. E. Grift, "Geometry-based Mass Grading of Mango Fruits using Image Processing," *Information Processing in Agriculture*, vol. 4, pp. 150–160, 2017.

[15] W. Aung, T. T. Thu, H. T. D. Aye, P. P. Htun and N. Z. Aung, "Weight Estimation of Mango from Single Visible Fruit Surface using Computer Vision," *2020 59th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, Chiang Mai, Thailand, 2020, pp. 366-371.

[16] R. Islamadina, N. Pramita, F. Arnia and K. Munadi, "Estimating fish weight based on visual captured," *2018 International Conference on Information and Communications Technology (ICOIACT)*, Yogyakarta, Indonesia, 2018, pp. 366-372.

[17] L. Xu, Y. Zhao, "Automated Strawberry Grading System based on Image Processing," *Computers and Electronics in Agriculture*, vol. 71, pp. 32–39, 2010.

[18] C. S. Nandi, B. Tudu and C. Koley, "Machine vision based automatic fruit grading system using fuzzy algorithm," *Proceedings of The 2014 International Conference on Control, Instrumentation, Energy and Communication (CIEC)*, Calcutta, India, 2014, pp. 26-30.

[19] I. Nyalala, C. Okinda, L. Nyalala, N. Makange, Q. Chao, L. Chao, K. Yousaf, and K. Chen, "Tomato volume and mass estimation using computer vision and machine learning algorithms: Cherry tomato model," *Journal of Food Engineering*, vol. 263, pp. 288–298, Dec. 2019.

[20] V. R. Phate, R. Malmathanraj and P. Palanisamy, "An Indirect Method to Estimate Sweet Lime Weight through Machine Learning Algorithm," *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, Erode, India, 2020, pp. 194-198.

[21] B. Zhang, N. Guo, J. Huang, B. Gu, J. Zhou, "Computer Vision Estimation of the Volume and Weight of Apples by Using 3D Reconstruction and Noncontact Measuring Methods," *Journal of Sensors*, vol. 2020, Article ID 5053407, 12 pages, 2020.

[22] F. S. A. Sa'ad, M. F. Ibrahim, A. M. Shakaff, A. Zakaria and M. Z. Abdullah, "Shape and Weight Grading of Mangoes using Visible Imaging," *Computers and Electronics in Agriculture*, vol. 115, pp. 51–56, 2015.

[23] A. Tafuro, A. Adewumi, S. Parsa, G. E. Amir and B. Debnath, "Strawberry picking point localization ripeness and weight estimation," *2022 International Conference on Robotics and Automation (ICRA)*, Philadelphia, PA, USA, 2022, pp. 2295-2302.

[24] D. Mengoli, G. Bortolotti, M. Piani and L. Manfrini, "On-line real-time fruit size estimation using a depth-camera sensor," *2022 IEEE Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, Perugia, Italy, 2022, pp. 86-90.

[25] D. K. Barbole and P. M. Jadhav, "Grape Yield Prediction using Deep Learning Regression Model," *2022 International Conference for Advancement in Technology (ICONAT)*, Goa, India, 2022, pp. 1-6.

[26] S. Suwannakhun and P. Daungmala, "Estimating Pig Weight with Digital Image Processing using Deep Learning," *2018 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, Las Palmas de Gran Canaria, Spain, 2018, pp. 320-326.

[27] C. B. Barber, D. P. Dobkin, H. Huhdanpaa, "The Quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469-83, Dec. 1996.

[28] M. Teichmann and M. Capps, "Surface reconstruction with anisotropic density-scaled alpha shapes," *Proceedings Visualization '98 (Cat. No.98CB36276)*, Research Triangle Park, NC, USA, 1998, pp. 67-72.

[29] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva and G. Taubin, "The ball-pivoting algorithm for surface reconstruction," in *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, no. 4, pp. 349-359, Oct.-Dec. 1999.