IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 12 November 2024, accepted 26 November 2024, date of publication 27 November 2024,
date of current version 17 December 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3507916

RESEARCH ARTICLE

Three-Dimensional Plant Model Development Through Image Recognition

YUYA HOSODA[1,2], (Member, IEEE), **GANZURKH BILGUUN**[3], **JIN OBOSHI**[3], AND **HITOSHI GOTO**[2,3,4]

[1]Graduate School of Engineering Science, Osaka University, Osaka 560-8531, Japan
[2]Center for IT-based Education, Toyohashi University of Technology, Aichi 441-8580, Japan
[3]Graduate School of Computer Science and Engineering, Toyohashi University of Technology, Aichi 441-8580, Japan
[4]Information and Media Center, Toyohashi University of Technology, Aichi 441-8580, Japan

Corresponding author: Yuya Hosoda (hosoda.yuya.es@osaka-u.ac.jp)

This work was supported by the Naito Research Grant.

ABSTRACT This paper proposes a novel method using image-recognition techniques to develop three-dimensional (3D) models of basil plants. Traditional approaches have dificulty scanning outdoor plants and stems overlapping outer leaves. In this paper, we collect 3D plant models in advance that reproduce the external and internal structures. Then, by selecting from the database the 3D model most similar to the actual plant in appearance, the proposed method develops 3D plant models using only images. However, collecting precise 3D models is a cost-intensive task. Based on the growth pattern that basil plants exhibit alternating leaves during growth, the proposed method automatically mass-produces realistic 3D plant models by assembling 3D leaf and stem models of actual plants. Additionally, we employ an image-recognition technique to extract embedding vectors from multi-angle images and assess the visual similarity between the actual plant and the realistic 3D plant model based on their cosine similarity. Finally, we construct a vector-search system incorporating k-means clustering and dimensionality reduction to limit the search scope and minimize computational complexity. Experimental results show that the proposed method efficiently obtains the most similar 3D model in the database, achieving a mean reciprocal rank of 0.90 and a search time of 0.003 s per query.

INDEX TERMS Computer vision, image recognition, plant, three-dimensional model, vector search.

I. INTRODUCTION

Three-dimensional (3D) models, digital data structures that represent actual objects in 3D space, have been utilized in various fields. In the healthcare sector [1], [2], 3D organ models have enabled early diagnosis and reduced surgical times. Geographic information systems have integrated 3D models of buildings and land for comprehensive spatial analyses in urban planning and environmental management [3], [4]. Compared with 2D images, 3D plant models of plants are more critical for elucidating plant phenotypes such as volume, shape, and posture [5], [6]. Additionally, replicas of actual plants offer a new direction for agricultural technology, enabling remote cultivation through digital twins [7], [8]. This paper aimed to develop 3D plant models that resemble actual plants to integrate the real and digital worlds seamlessly. However, developing accurate and realistic 3D plant models remains a significant challenge because their complex structures and diversity make it difficult to reproduce them fully. Data scarcity is also particularly pronounced for rare species and plants with long cultivation periods. These issues pose a significant barrier to plant phenotype research, remote agriculture, and the realization of digital twins. Therefore, there is a demand for developing more efficient and versatile methods for developing 3D plant models.

There are two main 3D scanning methods for acquiring 3D plant models: active and passive [9], [10]. Active methods involve measuring the reflections of light or lasers emitted by a transmitter to determine their distance from objects. Conversely, passive methods use the triangulation principle to obtain distance information based on the correspondence between multiple images. Although they can obtain high-precision 3D models, these methods require specialized

The associate editor coordinating the review of this manuscript and approving it for publication was Jiachen Yang.

equipment and scanning spaces, making them impractical for developing 3D models of outdoor plants. While the outer surface can be accurately reproduced, reproducing internal structures is also challenging due to overlapping outer leaves.

The development of alternatives to traditional active and passive 3D scanning technologies has gained momentum in recent years. Computer-aided design (CAD)-based methods [11] enable the creation of high-precision 3D plant models but require significant time, expertise, and the involvement of skilled professionals, making them less practical for large-scale applications. Deep learning has emerged as a promising tool for 3D model development, leveraging large-scale datasets to create models from textual descriptions [12], [13], a limited number of images [14], [15], or even a single image [16], [17]. However, these approaches heavily depend on the availability of specialized and expansive datasets. Distinct and varied morphologies further complicate this process, necessitating dedicated plant-specific datasets for practical model training. The development of high-precision 3D plant models still needs to be solved, particularly in the face of ecological diversity and limited datasets.

This paper introduces a novel approach to developing 3D plant models byutilizing visual similarity between actual plants and pre-constructed database models rather than striving for replication as in traditional methods. The proposed method effectively addresses challenges such as the occlusion of internal structures and the high cost associated with data acquisition through a component-based assembly process and a vector-search system leveraging image recognition models. Unlike photogrammetry [18], [19], [20], [21], which relies on numerous overlapping images from multiple viewpoints to reconstruct 3D models, our approach simplifies the process by utilizing only eight side-view images to approximate the plant structure. By prioritizing efficiency, scalability, and suitability for outdoor environments, the proposed method is well-suited for practical applications such as integration into remote cultivation systems for large-scale agriculture. Additionally, the proposed method thoughtfully balances computational efficiency and model quality for the use of the actual apprication. The key contributions of our method can be summarized as follows:

- We introduced an image recognition-based 3D plant model development method that eliminates the need for specialized equipment or complex scanning procedures. This approach overcomes the limitations of traditional active and passive scanning methods, enabling practical and efficient applications in outdoor environments.
- We developed a dataset comprising 4,096 realistic 3D plant models using 3D leaf and stem models of 16 sweet basil plants, followed by distinct alternating leaf growth patterns. If some components, such as leaves and stems, are scanned, this approach can seamlessly reproduce the actual plant structure in a 3D plant model. It facilitates the rapid expansion of datasets, even for rare plants with limited data, effectively addressing data scarcity in training generative models.
- We implemented a vector-search system to optimize the efficiency of navigating candidate 3D plant models while minimizing computational overhead. The system significantly enhances computational performance by integrating dimensionality reduction through singular value decomposition (SVD) and search scope narrowing via k-means clustering. This innovation facilitates the rapid development of 3D plant models across various applications, including digital twinning for large-scale agricultural farms.

The remainder of this paper is organized as follows. Section II introduces related works for 3D scanning techniques. Section III presents the cultivation environments and proposes a vector-search system using an image-recognition technique. Section IV discusses the experimental results, and Section V concludes the study.

A. 3D PLANT MODEL DEVELOPMENT

II. RELATED WORKS

Light-scanning approaches involve projecting structured light onto an object to capture surface and distance information by analyzing the time, angle, and distortion of reflected light [22], [23]. Although compact scanning devices can produce high-resolution 3D plant models, they require a controlled scanning environment conducive to stable light reflections, such as a dark room, which renders them impractical for scanning outdoor plants. Moreover, manually adjusting the light source position based on the object size increases the time required to generate 3D plant models.

Terrestrial laser scanning involves projecting laser light onto objects to acquire 3D information from the reflected light, making it suitable for outdoor environments. Light detection and ranging (LiDAR) is a prominent terrestrial laser-scanning technology that can scan objects over a wide area in real-time and has enabled non-destructive phenotype measurements of multiple vegetables [24], [25] and trees [26], [27]. However, it requires specialized equipment with high scalability costs, and the resolution of 3D models for small plants is limited. Additionally, the 3D models are represented by point clouds and do not include additional information such as color and texture.

The RGB-D camera method simultaneously captures color and depth information of objects through color and distance sensors [28], [29]. This information is integrated to generate 3D plant models by iteratively updating the correspondences between point clouds to minimize inter-point distances. However, it cannot acquire accurate depth information based on the reflectivity or transparency of the object surface, resulting in imprecise 3D plant models. Therefore, additional image processing is necessary to enhance the depth accuracy and mitigate model deficiencies.

Photogrammetry reconstructs 3D plant models by analyzing the geometric properties and visual features from multiple images captured from different angles.

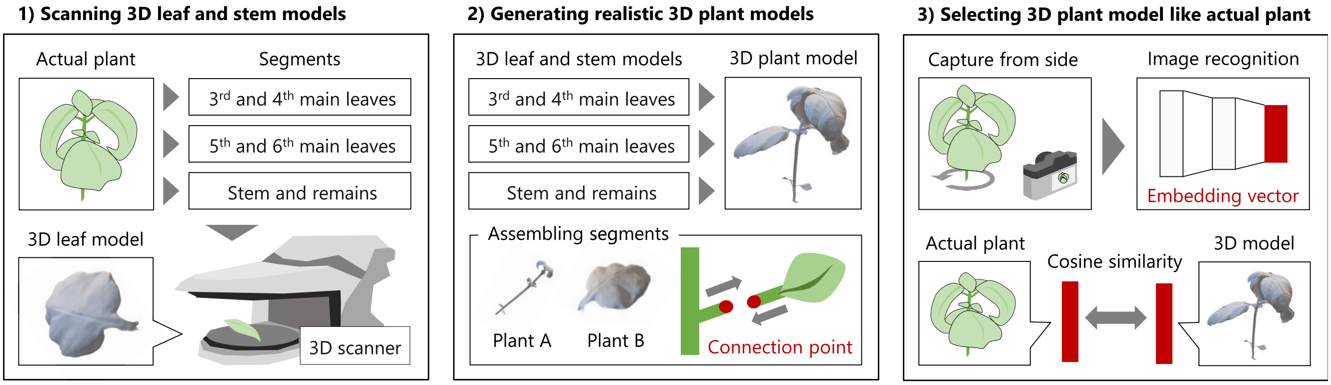

FIGURE 1. Development process of the proposed 3D plant modeling methodology.

The structure-from-motion technique calculates the camera position and orientation based on the correspondence between image characteristics [18], [19]. Multiview stereo techniques employ stereo matching across multiple viewpoints to compute the depth and surface normals, which are fused to generate a dense 3D point cloud [20], [21]. They enable the accurate reconstruction of minute plant components, provided that precise camera positions are available. However, occlusions pose challenges to capturing internal structures such as plant stems. Additionally, successful photogrammetric reconstructions require capturing many overlapping images under uniform lighting conditions.

Based on the limitations identified in traditional 3D scanning techniques, it is impractical to develop 3D plant models accurately. Therefore, this paper aims to select visually similar models from a database as substitutes rather than replicating them accurately. The proposed approach compares the appearance of 3D plant models with actual plants using a few images, which offers the advantage of generating 3D models of outdoor plants without specialized equipment. However, collecting many precise 3D plant models in advance is necessary. Our strategy consists of assembling prescanned individual components of actual plants to mass-produce realistic 3D plant models artificially. Although this paper focused on small plants, the proposed method can be applied to plants of any size and shows promise for broader applications under appropriate image-capture environments. Further details of the proposed method and the experimental design are presented in subsequent sections.

III. MATERIALS AND METHODS

Figure 1 depicts the development process of the proposed 3D plant modeling methodology. The process starts with segmenting an actual plant into discrete components, such as leaves and stems, which are then scanned to create detailed 3D models of each part. A realistic 3D plant model can be assembled by aligning and connecting these individual models. The final stage involves selecting the 3D plant model from the database that resembles the actual plant's appearance. First, by rotating around their stems, this approach captures the 3D plant model and the actual plant from the sides. It extracts embedding vectors from these images using an image recognition model and then calculates their cosine similarity to quantify the visual similarity.

A. CULTIVATION ENVIRONMENTS

This paper employed sweet basil plants to develop 3D plant models owing to their simple structure wherein the leaves grow in an alternating pattern. We cultivated 26 plants using a hydroponic cultivation kit (GS1 Max; JustSmart) in two batches. The first batch included six plants cultivated from September 23 to October 23, 2023, followed by the second batch including 20 plants from October 23 to November 22, 2023. The hydroponic cultivation kit was installed with LED lights at a height of 0.22 m. The air-temperature of the controlled indoor environment was maintained at 20–23 °C. The fertilizer temperature was 20–22 °C, with an electrical conductivity of 1.1–1.4 $\mathrm{mS\,cm^{-1}}$. Artificial lighting was provided for 16 h per day at a photosynthetic photon flux density of 250–300 $\mathrm{\mu mol\,m^{-2}\,s^{-1}}$. After a two-day germination period in the dark, liquid fertilizer was introduced on the seventh day. The plants with eight leaves were scanned on the final day using the structured light-scanning technique.

We employed a 3D scanner (VL-500; Keyence), which is ideal for creating precise 3D models of small and delicate objects. The 3D scanner is notable for its accuracy of up to 2 µm, substantial scanning area of 500 × 500 × 500 mm, and the capability to process data rapidly at a rate of 100,000 points/s. However, direct scanning of the plants failed to capture the internal structures, such as stems, because the leaves occluded them. To overcome this challenge, we scanned the plants by segmenting them into three parts: third and fourth main leaves, fifth and sixth main leaves, and the remaining parts. Each segment was carefully mounted on a pedestal and secured using clip stands, enabling scanning from 12 distinct angles. Consequently, we obtained mesh models accurately representing the intricate 3D structures of plants without color and texture information.

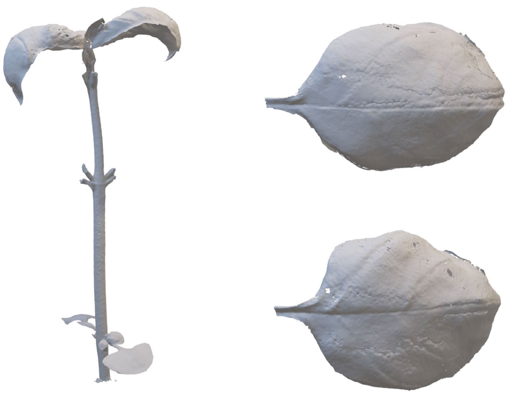

FIGURE 2. 3D leaf and stem models of the sweet basil plant.

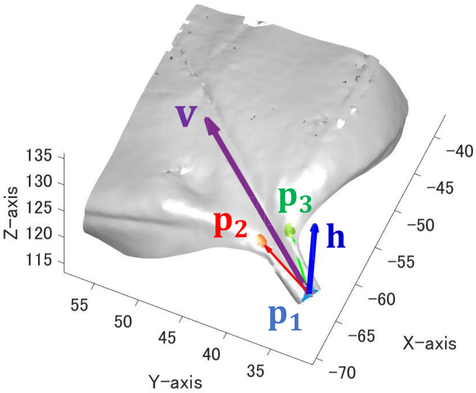

FIGURE 3. Vectors of the 3D plant parts.

The scanned mesh models included extraneous elements, such as clip stands, and exhibited surface holes in areas where the reflective light was not captured. Additionally, their large file sizes increased both the storage and computational complexity. Therefore, we employed MeshLab [30], a 3D processing software, to address these challenges. The postprocessing began with the manual removal of unwanted parts and filling of the surface holes. Subsequently, the file sizes were reduced by simplifying the mesh using a 3D grid with a cell size of 25 % to balance accuracy and resource efficiency. The final step involved surface smoothing to eliminate any remaining flaws. Figure 2 shows the 3D models of the leaves and stems of a sweet basil plant, illustrating that even intricate details such as branches were distinctly captured. Finally, we assembled these 3D models to generate realistic 3D plant models.

1) GENERATION OF REALISTIC 3D PLANT MODELS

The proposed method automatically generates realistic 3D plant models by assembling the 3D leaf and stem models to match the branches. In this paper, we define "*3D plant parts*" as 3D leaf and stem models with vectors representing the branch orientation and both leaf sides. Figure 3 shows the vectors of these 3D plant-part models. First, three coordinates on the branch are selected through a user-friendly interface using click functionality. These coordinates define the vectors; however, smaller branches can cause displacement errors. Therefore, the proposed method extends the specified coordinates to N nearest points. The center of gravity of the mesh formed using these points improves robustness against displacement errors. Let $\mathbf{p}_n (n = 1, 2, 3)$ denote the center of gravity of the nearest points. $\mathbf{p}_1$ corresponds to the edge of the branch and connection points and $\mathbf{p}_2$ and $\mathbf{p}_3$ are located on either sides at the end of the branch. Vector $\mathbf{v}$ indicates the branch orientation and is defined as follows:

$$\mathbf{v} = \frac{\mathbf{p}_2 + \mathbf{p}_3}{2} - \mathbf{p}_1. \tag{1}$$

We also defined the following normal vector representing both leaf sides:

$$\mathbf{h} = (\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_3 - \mathbf{p}_1), \tag{2}$$

where "×" denotes the cross-product operator.

The 3D plant parts are assembled to match the vectors at their connection points. As they comprise different coordinate axes, the proposed method adjusts them by using Rodriguez's rotation formula. Let the line and normal vectors of the two 3D plant parts be $\mathbf{v}^{\mathrm{A}}, \mathbf{h}^{\mathrm{A}}$ and $\mathbf{v}^{\mathrm{B}}, \mathbf{h}^{\mathrm{B}}$, respectively. The rotation axis $\mathbf{r}$ and rotation angle θ of the normal vector are obtained as

$$\mathbf{r} = \frac{\mathbf{h}^{\mathrm{A}} \times \mathbf{h}^{\mathrm{B}}}{|\mathbf{h}^{\mathrm{A}} \times \mathbf{h}^{\mathrm{B}}|}, \tag{3}$$

$$\theta = \arccos\left(\frac{\mathbf{h}^{\mathrm{A}} \cdot \mathbf{h}^{\mathrm{B}}}{|\mathbf{h}^{\mathrm{A}}||\mathbf{h}^{\mathrm{B}}|}\right), \tag{4}$$

where$|h|$ denotes the normal of h. Using $\mathbf{r} = [r_{\mathrm{x}}, r_{\mathrm{y}}, r_{\mathrm{z}}]^{\mathrm{T}}$, the rotation matrix for the normal vector $\mathbf{R}^{\mathrm{h}}$ is defined as

$$\mathbf{R}^{\mathrm{h}} = \mathbf{I} + \sin\theta\,\mathbf{K} + (1 - \cos\theta)\mathbf{K}^2, \tag{5}$$

where $\mathbf{I}$ denotes an identity matrix and

$$\mathbf{K} = \begin{bmatrix} 0 & -r_{\mathrm{z}} & r_{\mathrm{y}} \\ r_{\mathrm{z}} & 0 & -r_{\mathrm{x}} \\ -r_{\mathrm{y}} & r_{\mathrm{x}} & 0 \end{bmatrix}. \tag{6}$$

The rotation matrix for the line vectors $\mathbf{R}^{\mathrm{v}}$ is also calculated using the same method. Using these rotation matrices, the rotation matrix $\mathbf{R}$ can be obtained as follows:

$$\mathbf{R} = \mathbf{R}^{\mathrm{h}}\mathbf{R}^{\mathrm{v}}. \tag{7}$$

Applying $\mathbf{R}$ to another matrix transforms it into the same coordinate axis.

Figure 4 shows the original and realistic 3D plant models. In the original 3D plant model, the stem on the underside of the leaves could not be depicted because it did not irradiate the structured light. By contrast, the realistic 3D plant model exhibits the internal 3D structure. Additionally, replacements with 3D plant parts can easily generate various 3D plant models. Therefore, the proposed method rapidly develops a database containing numerous realistic 3D plant models.

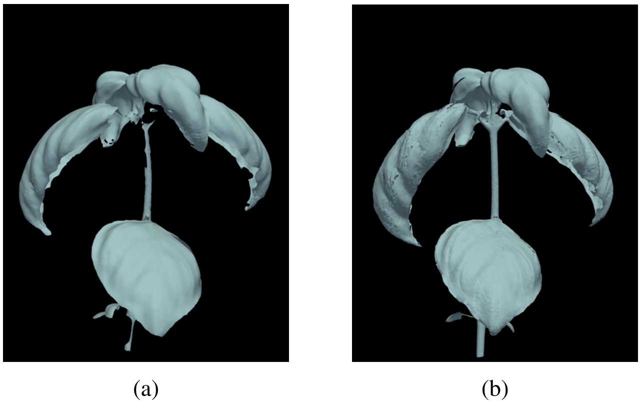

FIGURE 4. 3D plant models: (a) Original and (b) realistic.

TABLE 1. Relationship between the number of images and capture angles.

The number of images	Capture angles
1	0
2	0, 180
4	0, 90, 180, 270
8	0, 45, 90, 135, 180, 225, 270, 315

2) VISUAL SIMILARITY BETWEEN ACTUAL PLANTS AND THEIR 3D MODELS

The proposed method searches for the realistic 3D plant models that resemble the actual plants from the database. We focused on the fact that the appearance of 3D plant models is similar to that of the scanned plant. This section uses CNN-based image recognition to examine the visual similarity between the actual plants and their 3D models.

We captured actual plants using the web camera (UCAM-C520FBBK; ELECOM), which was positioned 30 cm away from the plant to ensure that it was in the center of the frame. Additionally, blackout curtains were installed behind the actual plant to ensure a consistent background. The plant was rotated around its stem axis at 45Â° increments. The "front" of the plant (0Â°) was determined as the position in which the first leaves opened to the left and right. Table 1 lists the capture angles of the N images used in this paper.

Similarly, images of the 3D plant model were captured from the side while rotating around the stem at 45Â° increments. The rendering engine was configured to ensure that the camera parameters produced images with consistent quality from all viewpoints in the actual camera settings. We also defined the front of the 3D model to be the same as that of the actual plant. However, unlike actual plants, the 3D plant models do not contain color and texture information. Hence, we employed grayscale images with a Gaussian filter to eliminate the texture information.

We extracted the embedding vectors from the images using three CNN-based image-recognition models: Visual Geometry Group (VGG16) [31], residual neural network (ResNet18) [32], and Inception-v3 [33]. The output of the final convolutional layers of these models encapsulates high-dimensional information about the image. We then concatenated their outputs, which were treated as embedding vectors. Let $\mathbf{f}^{\mathrm{A}}$ and $\mathbf{f}^{\mathrm{M}}$ denote the embedding vectors of the actual plant and its 3D model, respectively. We evaluate the visual similarity S using the cosine similarity between the embedding vectors as follows:

$$S = \frac{\mathbf{f}^{\mathrm{A}} \cdot \mathbf{f}^{\mathrm{M}}}{||\mathbf{f}^{\mathrm{A}}||\ ||\mathbf{f}^{\mathrm{M}}||}. \tag{8}$$

The closer the value of S is to 1, the more visually similar the actual plant and its 3D plant model, and vice versa.

TABLE 2. Comparisons of the visual similarity between the actual plants and 3D plant models of five sweet basil plants. The underlined values indicate the maximum similarity for each plant.

		3D plant model				
		Plant A	Plant B	Plant C	Plant D	Plant E
Actual plant	Plant A	$\underline{0.86}$	0.85	0.83	0.80	0.80
	Plant B	0.82	$\underline{0.86}$	0.80	0.80	0.79
	Plant C	0.82	0.81	$\underline{0.84}$	0.80	0.78
	Plant D	0.81	0.82	0.81	$\underline{0.84}$	0.82
	Plant E	0.77	0.79	0.79	0.79	$\underline{0.83}$

Table 2 presents the visual similarity values of five sweet basil plants with their 3D models. The overall visual similarity values are below 0.87 because the camera environments were not exactly similar, and the rotating axis was not always ensured to be at the stem of the actual plant. The diagonal components identified the highest values for combinations that targeted the same plant. These results indicate that 3D plant models resembling actual plants can be selected by verifying their appearance from multiple directions. The following section describes the development of a vector-search system based on the visual similarity to navigate candidate 3D plant models that closely resemble the actual plants.

3) VECTOR-SEARCH SYSTEM BASED ON VISUAL SIMILARITY

We develop a vector-search system to navigate candidate 3D plant models based on visual similarity; however, it has two problems. First, to represent the detailed 3D information about the plant, a long embedding vector extracted from multiangle images is required, leading to increased computational complexity and memory consumption. For example, the dimensionality of the embedding vector extracted from eight images using VGG16 reaches 32,768. Second, the model-search time is proportional to the number of 3D plant models; however, an extensive database contains 3D plant models that more closely resemble actual plants. Therefore, we employ dimensionality reduction for the embedding vector and k-means clustering to limit the search scope.

Figure 5a shows a block diagram outlining the system-construction process. Let Q represent the output length of the image-recognition model. The proposed method extracts outputs from N images of 3D plant models and then combines them to create an embedding vector of length QN. We construct matrix $\mathbf{D} \in \mathbb{R}^{M \times QN}$ by concatenating these embedding vectors with M candidate 3D plant models.

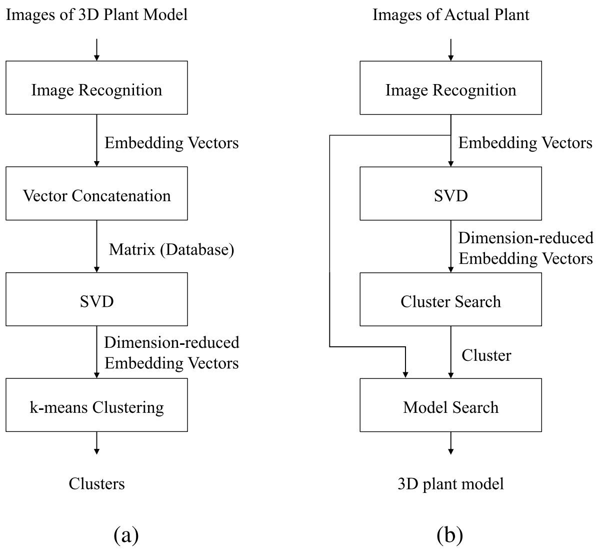

FIGURE 5. Block diagram of the proposed vector-search system. (a) System construction. (b) Model search.

The matrix functions as a database; however, its computational complexity and memory consumption increase with size. These embedding vectors are sparse and include patterns relevant to the plants. The proposed method employs SVD and k-means clustering to reduce the matrix size.

SVD breaks down matrix $\mathbf{D}$ into three key matrices: $\mathbf{U}$, Σ, and $\mathbf{V}$. Let T denote the number of singular values. Matrix $\mathbf{U} \in \mathbb{R}^{M \times T}$ contains the left singular vectors encapsulating the principal components of the rows. The diagonal matrix $\Sigma \in \mathbb{R}^{T \times T}$ contains singular values that quantify the importance of each principal component in $\mathbf{U}$ and $\mathbf{V}$. Meanwhile, the right singular matrix $\mathbf{V} \in \mathbb{R}^{QN \times T}$ comprises right singular vectors outlining the principal components of the columns. $\mathbf{D}$ can be reconstructed using $\mathbf{D} \approx \mathbf{U}\Sigma\mathbf{V}^{\mathrm{T}}$. In this paper, we approximate the matrix $\hat{\mathbf{D}} = \mathbf{U}\Sigma$ to reduce the row dimensionality. By retaining limited singular values, each row of $\hat{\mathbf{D}}$ corresponds to a dimension-reduced embedding vector that concisely captured the visual features of the plant.

We use the k-means clustering algorithm to narrow the search scope by leveraging the dimension-reduced embedding vectors. The algorithm randomly selects centroids by sorting the data into C unique clusters. Each data point is assigned to its closest centroid based on the cosine similarity between the dimension-reduced embedding vectors, which serve as a proximity measure. Subsequently, the centroids are updated to reflect the average positions of all points within their clusters. The cycle of assigning data points and updating centroids continues until the centroids remain constant or a predetermined number of iterations is reached.

Figure 5b presents a block diagram illustrating the search process for identifying the most suitable 3D plant model from the database. Initially, the proposed method extracts the embedding vector $\mathbf{d}$ from multiangle images of the actual plant. The dimension-reduced embedding vector is obtained using $\hat{\mathbf{d}} = \mathbf{dV}$ to reduce the computational complexity. The proposed method compares it with the centroids of each cluster by using cosine similarity to determine the best match. Subsequently, the candidate 3D plant model with the highest cosine similarity, which most closely resembles the actual plant, is searched within the cluster by employing the cosine similarity between the original embedding vectors.

IV. EXPERIMENTS AND DISCUSSION

A. EXPERIMENTAL CONDITIONS

In this paper, we extracted embedding vectors from images using the VGG16 [31], ResNet18 [32], and Inception-V3 [33] image-recognition models on Python 3.10. These models were trained using ImageNet [34]. The input-image size for VGG16 and ResNet18 was 224 × 224, and that for Inception-V3 was 250 × 250. The output lengths of VGG16, ResNet18, and Inception-V3 were $Q = 4096$, $Q = 512$, and $Q = 2048$, respectively.

The proposed method targeted 4,096 realistic 3D plant models assembled using 3D plant parts of 16 actual plants. We evaluated the performance and computational complexity by using $Q = 10$ actual plants whose 3D plant parts were not captured. The search performance of the model was evaluated using the mean reciprocal rank (MRR) and mean average precision (mAP) metrics. MRR indicates the average of the reciprocal ranks at which the first relevant item is found across a series of queries and is calculated as follows:

$$\mathrm{MRR} = \frac{1}{Q}\sum_{i=1}^{Q}\frac{1}{r_i}, \tag{9}$$

where r_i denotes the highest rank of the relevant item in the i-th query. The total number of search results was set to $L = 25$. The five 3D plant model candidates were treated as relevant items (correct models) in the experiment. The models were ranked in the top five in terms of their visual similarity to the 3D plant models of the actual plant scanned without partwise disassembly. The closer the visual similarity between the actual plant and any top-ranking correct model, the closer the MRR is to one. Additionally, mAP indicates the mean of the average precision (AP) across multiple queries. AP is defined as the proportion of relevant items in the search results and computed as follows:

$$\mathrm{AP} = \frac{\sum_{l=1}^{L} P(l) \cdot \mathrm{rel}(l)}{L}, \tag{10}$$

where $P(l)$ indicates the precision of the top l-th item in the search results and $\mathrm{rel}(l)$ is a function that returns 1 when the l-th item is relevant and 0 otherwise. As the proportion of correct models in the search results increased, the mAP approached 1. Regarding the computational complexity, we measured the time required to construct the vector-search system after extracting the embedding vectors from images (CONST) and searching for candidate 3D plant models per query (SEARCH). The vector-search system was designed using MATLAB 2023b.

TABLE 3. Performance evaluation of the vector-search system using ResNet18 (RES), VGG16 (VGG), and Inception-v3 (INC) models. We set $C = 16$ and $N = 8$.

T	Model	MRR	mAP	CONST [s]	SEARCH [s]
4096	RES	0.900	0.460	5.297	0.007
	VGG	0.900	0.440	5.489	0.050
	INC	0.900	0.400	5.161	0.024
2048	RES	0.900	0.460	2.554	0.007
	VGG	0.900	0.440	2.612	0.049
	INC	0.900	0.380	2.455	0.026
1024	RES	0.900	0.460	1.243	0.007
	VGG	0.900	0.420	1.258	0.050
	INC	0.900	0.380	1.172	0.028

We also evaluated the structural similarity to compare the weights of the plants and the highest-ranking 3D plant models in the search results. The fresh weights of the actual plants, excluding the roots, were measured. In contrast, the weights of the 3D plant models were calculated by adding the fresh weights of each plant part. We used two metrics to assess the errors: mean squared error (MSE) to measure the absolute error and mean absolute percentage error (MAPE) to measure the relative error. When the structure of a 3D plant model is similar to that of an actual plant, the difference in their weights should be slight, and the values of these metrics should approach 0.

B. EXPERIMENTAL RESULTS AND DISCUSSION

Table 3 presents the performance evaluation results of the vector-search systems employing the three image-recognition models. All models achieved an MRR of 0.900, indicating that correct models frequently appeared at the top of the search results. Regarding the mAP, the ResNet18-based approach consistently outperformed the other approaches across all dimension-reduced embedding vector lengths. This superior performance was attributed to the shortest output lengths obtained from the final convolutional layer of ResNet18, suggesting that it was less susceptible to information loss during dimensionality reduction. However, the mAP did not exceed 0.46, indicating that the search results did not include all correct models. Because all 3D plant models represent specific plants, it is difficult to completely distinguish the correct model from others based on visual similarities.

Shorter dimensionality-reduced embedding vectors correlate with lower system construction times because the calculation of the cosine similarity between dimension-reduced embedding vectors accounts for most of the clustering processes. However, this has a negligible effect on the model-search times because the original embedding vectors are used to search the models within the clusters. The ResNet18-based approach achieved the lowest search times, enabling image representation with shorter embedding vectors, and is practical for searching 3D plant models based on visual similarity.

The performance results of the vector-search system using ResNet18 to extract the embedding vectors from the images are presented in Table 4. The MRR and mAP decreased as T decreased. Because dimensionality reduction reduces the image information in embedding vectors, distinguishing between the candidate 3D plant models becomes more difficult. The performance drop is mitigated for $N = 4, 8$ because comprehensive information regarding the 3D structure of the plant can be extracted from multiple images. Furthermore, the system-construction time was significantly reduced by over 90 % when comparing $T = 4096$ with $T = 256$, highlighting the importance of reducing computational complexity via dimensionality reduction of the database. The model-search time remained relatively unaffected by T. Almost all the computations were allocated to searching for models within the cluster, wherein the calculation time was proportional to the length of the original embedding vector. Thus, the model-search performance and the system construction time were a trade-off according to T.

An increase in N led to improvements in the MRR and mAP. The images captured from the front ($N = 1, 2$) with $C = 16$ exhibited an MRR of less than 0.25, whereas those captured from eight directions ($N = 8$) exhibited an MRR greater than 0.80. Thus, capturing images from various angles is necessary to represent 3D structures using embedding vectors effectively. However, increasing the number of images resulted in longer embedding vectors, extending both the system construction and model-search times. Thus, N also offers a trade-off between computational complexity and model-search performance.

An increase in C diminished the model-search performance. The correct model could have been absent from the selected cluster because there were fewer candidate 3D plant models per cluster. Despite using the images captured from eight directions, the mAP decreased by 0.360 when comparing $C = 32$ with $C = 4$. Embedding vectors must include information on 3D structures from multiple perspectives beyond horizontal angles. An MRR of 0.900 was achieved at $C = 16$, indicating that the correct models were among the top search results within their clusters. Therefore, the proposed method selected candidate 3D plant models within a cluster resembling actual plants.

Without employing dimensionality reduction ($T = 4096$), the system construction time for $C = 32$ was approximately 1.5 times longer than that for $C = 4$. The k-means clustering algorithm updates the centroids of each cluster to generate clusters that better represent the data points within the dataset, and the cluster quantity affects the system construction time. Nevertheless, the impact of the cluster quantity on the system construction time was reduced with T because the process primarily involved cosine similarity calculations. Additionally, the model-search time was reduced by over 85 % for $C = 32$ compared with $C = 4$ by searching with embedding vectors derived from eight images. Therefore, the search-scope limitation using the clustering approach expedited the efficient search performance.

Figure 6 depicts the weight distributions of 3D plant models and actual plants. The 3D plant model corresponds to that in highest ranked search result of the vector-search system. The weights of the 3D plant models did not

TABLE 4. Performance evaluation of the vector-search system across three metrics. The embedding vectors were extracted from the images using ResNet16. (a) MRR. (b) mAP. (c) CONST [s]. (d) SEARCH [s].

		$T =$ 4096	$T =$ 2048	$T =$ 1024	$T =$ 512	$T =$ 256
C=4	N=1	0.108	0.108	0.108	0.083	0.083
	N=2	0.322	0.322	0.322	0.322	0.322
	N=4	0.575	0.575	0.575	0.575	0.575
	N=8	1.000	1.000	1.000	1.000	1.000
C=8	N=1	0.178	0.178	0.178	0.063	0.078
	N=2	0.291	0.283	0.291	0.283	0.267
	N=4	0.350	0.350	0.350	0.350	0.350
	N=8	0.900	0.900	0.900	0.900	0.900
C=16	N=1	0.095	0.015	0.015	0.040	0.045
	N=2	0.242	0.242	0.242	0.242	0.242
	N=4	0.733	0.733	0.733	0.733	0.733
	N=8	0.900	0.900	0.900	0.900	0.900
C=32	N=1	0.167	0.167	0.156	0.133	0.153
	N=2	0.270	0.267	0.267	0.308	0.262
	N=4	0.725	0.675	0.670	0.670	0.650
	N=8	0.800	0.800	0.800	0.800	0.800

(a)

		$T =$ 4096	$T =$ 2048	$T =$ 1024	$T =$ 512	$T =$ 256
C=4	N=1	0.025	0.025	0.025	0.020	0.020
	N=2	0.124	0.124	0.124	0.124	0.124
	N=4	0.263	0.263	0.263	0.264	0.270
	N=8	0.820	0.820	0.820	0.820	0.800
C=8	N=1	0.045	0.045	0.045	0.017	0.021
	N=2	0.104	0.100	0.104	0.101	0.101
	N=4	0.195	0.195	0.195	0.195	0.195
	N=8	0.520	0.520	0.520	0.520	0.500
C=16	N=1	0.019	0.003	0.003	0.008	0.011
	N=2	0.072	0.072	0.072	0.072	0.072
	N=4	0.319	0.320	0.320	0.320	0.309
	N=8	0.460	0.460	0.460	0.460	0.520
C=32	N=1	0.044	0.044	0.039	0.039	0.038
	N=2	0.086	0.106	0.085	0.099	0.112
	N=4	0.296	0.268	0.254	0.256	0.271
	N=8	0.460	0.460	0.460	0.460	0.460

(b)

		$T =$ 4096	$T =$ 2048	$T =$ 1024	$T =$ 512	$T =$ 256
C=4	N=1	0.463	0.231	0.102	0.040	0.028
	N=2	1.089	0.475	0.223	0.109	0.040
	N=4	2.200	1.043	0.460	0.215	0.111
	N=8	4.359	2.164	1.033	0.469	0.218
C=8	N=1	0.554	0.229	0.096	0.043	0.027
	N=2	1.271	0.542	0.241	0.079	0.041
	N=4	2.642	1.219	0.532	0.212	0.076
	N=8	4.898	2.631	1.227	0.521	0.212
C=16	N=1	0.605	0.216	0.105	0.056	0.040
	N=2	1.361	0.535	0.198	0.090	0.050
	N=4	2.733	1.315	0.531	0.180	0.082
	N=8	5.297	2.554	1.243	0.523	0.157
C=32	N=1	0.585	0.249	0.146	0.092	0.068
	N=2	1.547	0.554	0.247	0.136	0.086
	N=4	3.177	1.453	0.523	0.229	0.120
	N=8	6.332	3.091	1.401	0.464	0.215

(c)

		$T =$ 4096	$T =$ 2048	$T =$ 1024	$T =$ 512	$T =$ 256
C=4	N=1	0.004	0.004	0.004	0.004	0.004
	N=2	0.012	0.007	0.007	0.007	0.007
	N=4	0.013	0.018	0.014	0.017	0.015
	N=8	0.030	0.027	0.027	0.028	0.031
C=8	N=1	0.003	0.005	0.003	0.003	0.003
	N=2	0.004	0.004	0.004	0.004	0.003
	N=4	0.009	0.008	0.008	0.011	0.008
	N=8	0.018	0.018	0.015	0.016	0.015
C=16	N=1	0.002	0.001	0.001	0.001	0.002
	N=2	0.002	0.002	0.002	0.002	0.002
	N=4	0.005	0.005	0.005	0.005	0.005
	N=8	0.007	0.007	0.007	0.008	0.008
C=32	N=1	0.001	0.001	0.001	0.001	0.001
	N=2	0.001	0.001	0.002	0.001	0.002
	N=4	0.003	0.003	0.003	0.003	0.003
	N=8	0.004	0.004	0.004	0.004	0.004

(d)

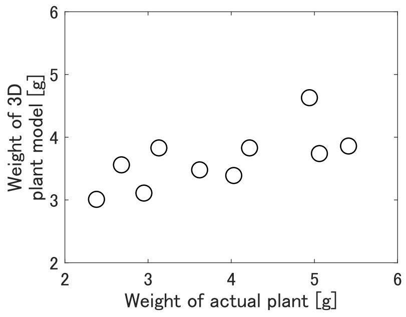

FIGURE 6. Weights of 3D plant models and actual plants. The 3D plant model corresponds to that in highest ranked search result of the vector-search system.

always align with those of actual plants. This discrepancy was attributed to the images being captured solely from horizontal angles, wherein the embedding vectors failed to include the vertical structures. The proposed method incorrectly categorized candidate 3D plant models that were similar in shape but differed in size. Additionally, as we obtained 3D plant parts from smaller plants in this paper, mostly small candidates were automatically developed by assembling them.

Table 5 presents the structural similarity results of the 3D plant models by comparing their weights with those of actual plants. Although the VGG16-based approach achieved the lowest root mean squared error (RMSE) of 0.915, the MAPE exceeded 21% across all approaches. The deviation of the 3D structure from the actual plants was independent of the image-recognition model employed. However, the RMSE decreased below 0.77 for smaller plants weighing less than 4.5 g. Because absolute errors are vulnerable to outliers, smaller plants were considered to have fewer divergences than larger ones. Thus, when appropriately sized 3D plant parts are available, 3D plant models with structures similar to those of actual plants can be developed. To improve the performance further, it is necessary to design embedding vectors that include size information and create an extensive database using 3D plant parts obtained from plants of various sizes. However, the larger the length of the embedding vector and the higher the number of 3D plant models, the greater the computational complexity of the vector-search system. Therefore, the dimensionality reduction and clustering techniques employed in this paper can effectively expand the proposed method.

C. DISCUSSION

The experimental results highlight the effectiveness of the proposed vector-search system in developing 3D plant

TABLE 5. Structure similarities of 3D plant models by comparing weights with actual plants across all plants (ALL) and those under 4.5 g (LOW). The vector-search system employs the following settings: $C = 16$, $N = 8$, and $T = 256$.

	RMSE		MAPE [%]	
Approach	ALL	LOW	ALL	LOW
ResNet18-based	0.967	0.620	21.323	18.039
VGG16-based	0.915	0.769	21.323	21.745
Inception-v3-based	1.244	0.716	22.575	15.837

models that closely approximate actual plants. Achieving an MRR of 0.90, the system reliably retrieves the most visually similar models from the pre-constructed database containing 4,096 realistic 3D plant models. With a search time of just 0.003 seconds per query, it underscores its computational efficiency, rendering it suitable for real-time applications. Key to this performance was integrating dimensionality reduction using SVD and k-means clustering to narrow search scopes, enhancing scalability and leading to significant computational savings, including a 90 % reduction in system construction time and a 75 % decrease in query search time.

Photogrammetry produces 3D models by reconstructing depth information from keypoint correspondences between overlapping images taken from various angles but demands increased computational resources and specialized equipment. For example, Paturkar et al.'s photogrammetry method [19] requires at least 17 minutes of post-processing per plant. Similarly, Feldmann et al. devised a method for measuring the 3D shape of fruits at low cost and high speed [21], but it requires meticulous calibration using specialized equipment and is unsuitable for 3D plant models with complex structures. In contrast, our method, which simplifies the modeling process by focusing on semantic features captured from a limited number of viewpoints, demonstrates that capturing images from just eight horizontal directions is sufficient to encapsulate the essential structural features of plants. Even a limited number of images can effectively facilitate structurally similar 3D plant models without requiring extensive computational resources or specialized equipment. This efficiency level aligns well with the demands of modern precision agriculture and environmental monitoring for commercial plant cultivation, where fast and reliable data processing is critical.

While the proposed vector-search system demonstrates significant promise in generating 3D plant models that closely resemble actual plants, there are inherent limitations to our current approach. A fundamental limitation lies in its reliance on images captured from horizontal viewpoints, which may overlook critical vertical and oblique structural details. This reliance introduces the potential for misidentification of 3D plant models that are structurally similar in horizontal aspects but differ in scale or vertical structure. Additionally, our method has only been validated on sweet basil plants with relatively simple and uniform structures. Since other plant species exhibit a wide variety of growth patterns and morphological complexities, the generalizability of our approach remains to be thoroughly evaluated.

Future work will enhance the embedding features by capturing images from multiple perspectives, including vertical and oblique angles. By incorporating positional information rather than merely combining their embedding vectors, we aim to represent the 3D structure of plants more effectively. This multi-view approach is expected to capture a more comprehensive set of semantic features, thereby improving the accuracy of structural similarity assessments and reducing the potential for misidentification. Additionally, we plan to build an expanded database that includes plants other than sweet basils. By handling plants at different growth stages, we can facilitate the development of 3D plant models across different scales and complexities. These improvements will enable a more precise 3D plant modeling system to handle diverse plant species and structures.

V. CONCLUSION

This paper proposed an efficient vector-search system that obtains models closely resembling actual plants based on visual similarity among 4,096 realistic 3D plant models. The experimental results showed that the top search results contained the correct models with an MRR of 0.90. Additionally, the system construction time was reduced by more than 80 % via dimensionality reduction, and the model-search time per query was decreased by 75 % using k-means clustering. Experiments on weight distribution confirmed that the proposed method could develop 3D plant models structurally similar to actual plants when appropriately sized 3D plant parts are available. In the future, we plan to design embedding vectors that can better distinguish 3D plant models by capturing images from angles other than the horizontal direction. Another future research direction is to develop an extensive database of 3D plant parts to represent the 3D structures of larger plants.

REFERENCES

[1] J. Meyer-Szary, M. S. Luis, S. Mikulski, A. Patel, F. Schulz, D. Tretiakow, J. Fercho, K. Jaguszewska, M. Frankiewicz, E. Pawłowska, R. Targoński, L. Szarpak, K. Dądela, R. Sabiniewicz, and J. Kwiatkowska, "The role of 3D printing in planning complex medical procedures and training of medical professionals—Cross-sectional multispecialty review," *Int. J. Environ. Res. Public Health*, vol. 19, no. 6, p. 3331, Mar. 2022.

[2] X. Zhang, E.-M. Otoo, Y. Fan, C. Tao, T. Wang, and K. Rhode, "Autostereoscopic 3D augmented reality navigation for laparoscopic surgery: A preliminary assessment," *IEEE Trans. Biomed. Eng.*, vol. 70, no. 4, pp. 1413–1421, Apr. 2023.

[3] I. Nishanbaev, "A web repository for geo-located 3D digital cultural heritage models," *Digit. Appl. Archaeol. Cultural Heritage*, vol. 16, Mar. 2020, Art. no. e00139.

[4] I. M. Badwi, H. M. Ellaithy, and H. E. Youssef, "3D-GIS parametric modelling for virtual urban simulation using CityEngine," *Ann. GIS*, vol. 28, no. 3, pp. 325–341, Jul. 2022.

[5] D. Li, G. Shi, W. Kong, S. Wang, and Y. Chen, "A leaf segmentation and phenotypic feature extraction framework for multiview stereo plant point clouds," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 2321–2336, 2020.

[6] D. Li, G. Shi, J. Li, Y. Chen, S. Zhang, S. Xiang, and S. Jin, "PlantNet: A dual-function point cloud segmentation network for multiple plant species," *ISPRS J. Photogramm. Remote Sens.*, vol. 184, pp. 243–263, Feb. 2022.

[7] Y. Zhang, P. Teng, M. Aono, Y. Shimizu, F. Hosoi, and K. Omasa, "3D monitoring for plant growth parameters in field with a single camera by multi-view approach," *J. Agricult. Meteorol.*, vol. 74, no. 4, pp. 129–139, 2018.

[8] H. Moreno and D. Andújar, "Proximal sensing for geometric characterization of vines: A review of the latest advances," *Comput. Electron. Agricult.*, vol. 210, Jul. 2023, Art. no. 107901.

[9] K. Bartol, D. Bojanic, T. Petkovic, and T. Pribanic, "A review of body measurement using 3D scanning," *IEEE Access*, vol. 9, pp. 67281–67301, 2021.

[10] A. Sabato, S. Dabetwar, N. N. Kulkarni, and G. Fortino, "Noncontact sensing techniques for AI-aided structural health monitoring: A systematic review," *IEEE Sensors J.*, vol. 23, no. 5, pp. 4672–4684, Mar. 2023.

[11] D. Cascaval, M. Shalah, P. Quinn, R. Bodik, M. Agrawala, and A. Schulz, "Differentiable 3D CAD programs for bidirectional editing," *Comput. Graph. Forum*, vol. 41, no. 2, pp. 309–323, May 2022.

[12] C.-H. Lin, J. Gao, L. Tang, T. Takikawa, X. Zeng, X. Huang, K. Kreis, S. Fidler, M.-Y. Liu, and T.-Y. Lin, "Magic3D: high-resolution Text-to-3D content creation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 300–309.

[13] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall, "Dreamfusion: Text-to-3D using 2D diffusion," in *Proc. 11th Int. Conf. Learn. Represent.*, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10313063/references#references

[14] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," *Commun. ACM*, vol. 65, no. 1, pp. 99–106, Jan. 2022.

[15] Z. Chen, T. Funkhouser, P. Hedman, and A. Tagliasacchi, "MobileNeRF: Exploiting the polygon rasterization pipeline for efficient neural field rendering on mobile architectures," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 16569–16578.

[16] X.-F. Han, H. Laga, and M. Bennamoun, "Image-based 3D object reconstruction: State-of-the-art and trends in the deep learning era," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1578–1604, May 2021.

[17] Q. Yu, C. Yang, and H. Wei, "Part-wise AtlasNet for 3D point cloud reconstruction from a single image," *Knowledge-Based Syst.*, vol. 242, Apr. 2022, Art. no. 108395.

[18] J. Iglhaut, C. Cabo, S. Puliti, L. Piermattei, J. O'Connor, and J. Rosette, "Structure from motion photogrammetry in forestry: A review," *Current Forestry Rep.*, vol. 5, no. 3, pp. 155–168, Sep. 2019.

[19] A. Paturkar, G. S. Gupta, and D. Bailey, "Non-destructive and cost-effective 3D plant growth monitoring system in outdoor conditions," *Multimedia Tools Appl.*, vol. 79, nos. 47–48, pp. 34955–34971, Dec. 2020.

[20] S. Wu, W. Wen, W. Gou, X. Lu, W. Zhang, C. Zheng, Z. Xiang, L. Chen, and X. Guo, "A miniaturized phenotyping platform for individual plants using multi-view stereo 3D reconstruction," *Frontiers Plant Sci.*, vol. 13, Aug. 2022, Art. no. 897746.

[21] M. J. Feldmann and A. Tabb, "Cost-effective, high-throughput phenotyping system for 3D reconstruction of fruit form," *Plant Phenome J.*, vol. 5, no. 1, Jan. 2022, Art. no. 20029.

[22] M. Gupta, A. Agrawal, A. Veeraraghavan, and S. G. Narasimhan, "Structured light 3D scanning in the presence of global illumination," in *Proc. CVPR*, Jun. 2011, pp. 713–720.

[23] Y. Zhang, D. L. Lau, and Y. Yu, "Causes and corrections for bimodal multi-path scanning with structured light," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4426–4434.

[24] D. Reiser, M. Vázquez-Arellano, D. S. Paraforos, M. Garrido-Izard, and H. W. Griepentrog, "Iterative individual plant clustering in maize with assembled 2D LiDAR data," *Comput. Ind.*, vol. 99, pp. 42–52, Aug. 2018.

[25] S. Jin, Y. Su, F. Wu, S. Pang, S. Gao, T. Hu, J. Liu, and Q. Guo, "Stem–leaf segmentation and phenotypic trait extraction of individual maize using terrestrial LiDAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1336–1346, Mar. 2019.

[26] I. Indirabai, M. V. H. Nair, R. Jaishanker, and R. Nidamanuri, "Terrestrial laser scanner based 3D reconstruction of trees and retrieval of leaf area index in a forest environment," *Ecolog. Informat.*, vol. 53, Jul. 2019, Art. no. 100986. [Online]. Available: https://www.mdpi.com/2075-5309/11/9/417

[27] W. Li, Z. Niu, R. Shang, Y. Qin, L. Wang, and H. Chen, "High-resolution mapping of forest canopy height using machine learning by coupling ICESat-2 LiDAR with sentinel-1, Sentinel-2 and Landsat-8 data," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 92, Oct. 2020, Art. no. 102163.

[28] T. Yang, J. Ye, S. Zhou, A. Xu, and J. Yin, "3D reconstruction method for tree seedlings based on point cloud self-registration," *Comput. Electron. Agricult.*, vol. 200, Sep. 2022, Art. no. 107210.

[29] B. Wei, "Dynamic simulation of leaf area index for the soybean canopy based on 3D reconstruction," *Ecol. Inform.*, vol. 75, Mar. 2023, Art. no. 102070. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574954124000311#bi0005

[30] P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, and G. Ranzuglia, "MeshLab: An open-source mesh processing tool," in *Proc. Eurographics Italian chapter Conf.*, Salerno, Italy, Jan. 2008, pp. 129–136.

[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent.*, 2015. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7913730/references#references

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[33] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[34] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

YUYA HOSODA (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Osaka University, Osaka, Japan, in 2017, 2019, and 2022, respectively. He was an Assistant Professor with Toyohashi University of Technology, Aichi, Toyohashi, from 2022 to 2024. Since 2024, he has been an Assistant Professor with the Graduate School of Engineering Science, Osaka University. His research interest includes image and speech signal processing.

GANZURKH BILGUUN received the B.E. degree from Toyohashi University of Technology, Aichi, Japan, in 2024. His research interest includes computer vision.

JIN OBOSHI received the B.E. degree from Toyohashi University of Technology, Aichi, Japan, in 2024. His research interest includes agricultural information.

HITOSHI GOTO received the B.E., M.E., and Ph.D. degrees from Hokkaido University, Hokkaido, Japan, in 1989, 1991, and 1993, respectively. He was a Research Associate with Tohoku University, Miyagi, Japan, from 1996 to 1998; and an Associate Professor with Toyohashi University of Technology, Toyohashi, Japan, from 1998 to 2020. Since 2020, he has been a Professor with the Information and Media Center, Toyohashi University of Technology. His research interests include computational chemistry and cheminformatics for organic compounds and materials.