Boosting grape bunch detection in RGB-D images using zero-shot annotation with Segment Anything and GroundingDINO

by Devanna, R.P., Reina, G., Cheein, F.A. and Milella, A.

Devanna, R.P., Reina, G., Cheein, F.A. and Milella, A. (2025) 'Boosting grape bunch detection in RGB-D images using zero-shot annotation with Segment Anything and GroundingDINO' *Computers and Electronics in Agriculture*, 229, article number 109611.

02 December 2024

Computers and Electronics in Agriculture 229 (2025) 109611

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

Boosting grape bunch detection in RGB-D images using zero-shot annotation with Segment Anything and GroundingDINO

Rosa Pia Devanna [a,*,1], Giulio Reina [b], Fernando Auat Cheein [c,d], Annalisa Milella [a]

[a] Institute of Intelligent Industrial Technologies and Systems for Advanced Manufacturing (STIIMA), National Research Council of Italy (CNR), via Amendola 122 D-O, Bari, 70126, Italy

[b] Department of Mechanics, Mathematics and Management, Polytechnic University of Bari, Via Orabona 4, Bari, 70125, Italy

[c] Department of Engineering, Harper Adams University, England, UK

[d] Department of Electronic Engineering, Advanced Center for Electrical and Electronic Engineering (AC3E), Federico Santa Maria Technical University, Valparaiso, Chile

ARTICLE INFO

Keywords:
Grape bunch detection
Instance segmentation
Zero-shot networks
Precision agriculture
Agriculture robotics

ABSTRACT

Latest advances in artificial intelligence, particularly in object recognition and segmentation, provide unprecedented opportunities for precision agriculture. This work investigates the use of state-of-the-art AI models, namely Meta's *Segment Anything* (SAM) and *GroundingDino*, for the task of grape cluster detection in vineyards. Three different methods aimed at enhancing the instance segmentation process are proposed: (i) SAM-Refine (SAM-R), which refines a previously proposed depth-based clustering approach, referred to as DepthSeg, using SAM; (ii) SAM-Segmentation (SAM-S), which integrates SAM with a pre-trained semantic segmentation model to improve cluster separation; and (iii) AutoSAM-Dino (ASD), which eliminates the need for manual labeling and transfer learning through the combined use of GroundingDino and SAM. Analysis is conducted on both the object counting and pixel-level segmentation accuracy against a manually labeled ground truth. Metrics such as mean Average Precision (mAP), Intersection over Union (IoU), and precision and recall are calculated to assess the system performance. Compared to the original DepthSeg algorithm, SAM-R slightly advances object counting (mAP: +0.5%) and excels in pixel-level segmentation (IoU: +17.0%). SAM-S, despite a mAP decrease, improves segmentation accuracy (IoU: +13.9%, Precision: +9.2%, Recall: +11.7%). Similarly, ASD, although with a lower mAP, shows significant accuracy enhancement (IoU: +7.8%, Precision: +4.2%, Recall: +4.9%). Additionally, from a labor effort point of view, instance segmentation techniques require much less time for training than manual labeling.

1. Introduction

The use of Artificial Intelligence (AI) has triggered a paradigm change in precision agriculture, with the aim of improving crop management to maximize yields while minimizing environmental impact (Eli-Chukwu, 2019; Saiz-Rubio and Rovira-Más, 2020). Specifically, the incorporation of deep learning into precision agriculture has led to significant advancements, demonstrating the extensive capability of this technology to improve crop management and analysis (Liu et al., 2023; Lin et al., 2019). Numerous breakthroughs have been made possible by the use of convolutional neural networks (CNNs), including complex methods for tasks like semantic segmentation (Ciarfuglia et al., 2023) and object recognition (Shen et al., 2023) within agricultural imagery. These techniques have facilitated automated high-throughput phenotyping, disease detection, and yield prediction, demonstrating the transformative impact of deep learning (Casado-García et al., 2022).

So far, the main disadvantage of deep learning is the need for massive datasets to train the large number of weights of the entire net from scratch. Another significant limitation is the amount and quality of labeled data. For the network to learn its prediction task, the data must be properly annotated, which is typically done manually using image labeling tools. Manual labeling is generally labor-intensive when using natural images, since scenes can be cluttered and difficult to interpret even for an expert user. This problem can be mitigated by pre-trained networks and transfer learning strategies, which allow knowledge from a similar domain to be transferred in order to perform new tasks (Sharma et al., 2020). This approach leverages pre-trained models on large datasets, fine-tuning them for specific agricultural

* Corresponding author.
E-mail address: rosapia.devanna@stiima.cnr.it (R.P. Devanna).

[1] R.P. Devanna is pursuing her PhD in AI for Environment and Agriculture at the University of Naples Federico II, Italy.

https://doi.org/10.1016/j.compag.2024.109611
Received 11 January 2024; Received in revised form 23 October 2024; Accepted 29 October 2024
Available online 2 December 2024

tasks (Devanna et al., 2022; Milella et al., 2019). While transfer learning has reduced the need for extensive data collection and labeling, it still requires a considerable amount of domain-specific data and computational resources for training (Najafabadi et al., 2015; Kamilaris and Prenafeta-Boldú, 2018).

Several studies have explored grape bunch detection using various deep learning models. For instance, Ghoury et al. (2019) employed Faster R-CNN for grape detection, achieving significant accuracy but requiring extensive labeled datasets for training. Shen et al. (2022) utilized Mask R-CNN for instance segmentation of grape clusters, demonstrating good performance but also highlighting the challenges associated with manual annotation of training data. Traditional machine learning approaches, such as Support Vector Machines (SVMs) and k-Nearest Neighbors (k-NN) classifiers, have also been applied but often lack the generalization capabilities of deep learning models (Boateng et al., 2020). These methods, while effective to some extent, emphasize the ongoing need for solutions that reduce the dependency on large, annotated datasets.

Recent advancements in zero-shot models aim to overcome these limitations by enhancing the generalization capabilities of deep learning for scene segmentation and object detection tasks, while significantly reducing the need for manual labeling. In the context of precision agriculture, several studies have begun to explore the use of zero-shot learning techniques. For example, Zhong et al. (2020) applied zero-shot learning for plant disease identification, enabling models to recognize diseases without requiring extensive labeled datasets. Similarly, Williams et al. (2024) utilized zero-shot object detection to identify leaves in crop fields, reducing the reliance on large annotated datasets. Further insights about zero-shot networks in precision agriculture can be found in Singh and Sanodiya (2023), Tan et al. (2024). Overall, this approach is particularly advantageous, as it eliminates the extensive labor and time required for dataset annotation (Liu, 2023; Zhang et al., 2023), which is a major bottleneck in deploying AI solutions in agriculture. By leveraging zero-shot models, we can achieve comparable or even superior segmentation performance without the exhaustive preparation typically associated with traditional deep learning approaches, Luo et al. (2016), Van Klompenburg et al. (2020).

Among zero-shot learning models, Meta's SegmentAnything Model (SAM) (Osco et al., 2023) has been demonstrated to be particularly promising for several applications. SAM is a high-quality object mask generator; it is a transformer-based model that has been trained on a large dataset of 11 million images and 1.1 billion masks. The model is designed to perform zero-shot learning, which means it can generate object masks for images it has never seen before during training. The model uses a transformer architecture (Lin et al., 2022), a type of deep learning model based on self-attention mechanisms that has been successfully adopted in various applications, including natural language processing (Kalyan et al., 2021) and image segmentation (Khan et al., 2022). The transformer architecture allows the model to handle long-range dependencies in the data and adapt to new tasks with minimal fine-tuning (Wang et al., 2023).

Complementary to SAM, GroundingDINO (Cheng et al., 2023) is an open-set object detector by IDEA-Research. This model introduces a novel approach to object detection by combining the strengths of the Transformer-based detector DINO (DIstillation of knowledge with No labels) (Zhang et al., 2022) with grounded pre-training (Gan et al., 2022), enabling it to detect arbitrary objects with human inputs such as category names or referring expressions. The model is based on a transformer architecture, similar to the SAM model. It introduces language to a closed-set detector for open-set concept generalization. To effectively fuse language and vision modalities, the model is divided into three phases: a feature enhancer, a language-guided query selection, and a cross-modality decoder for cross-modality fusion. The GroundingDINO model has shown impressive performance in both zero-shot and fine-tuned scenarios. It performs remarkably well on benchmarks such as COCO, LVIS, ODinW, and RefCOCO/+/g2. It sets a new record on the ODinW zero-shot benchmark with a mean 26.1 AP.

Research presented in this work aims at evaluating the potential of zero-shot networks for the task of grape bunch detection and counting in precision viticulture. In previous work by the authors (Devanna et al., 2023), a novel approach to grape bunch detection and counting, hereinafter referred to as DepthSeg, was proposed. It combines semantic segmentation with a depth-based clustering algorithm. DepthSeg uses visual and depth data provided by an RGB-D sensor for vineyard image processing. The RGB-D sensor captures vineyard images, which are processed using a pre-trained deep learning segmentation network to separate fruit from non-fruit regions. Transfer learning is applied to fine-tune the network using manually labeled field images. After isolating the grape regions through semantic segmentation, the algorithm employs depth data to separate individual bunches by detecting significant depth gradient changes. This also allows for the separation of adjacent grape clusters by exploiting depth discontinuities among grape bunch boundaries.

In this work, the DepthSeg method is compared against zero-shot AI models (Wang et al., 2019) and specifically Meta's SegmentAnything Model (SAM) (Kirillov et al., 2023) and GroundingDINO (Liu et al., 2023). To the best of our knowledge, this is the first work using SAM and GroundingDINO for the task of automated grape recognition in the field. Three different methods are proposed: (i) SAM-Refine (SAM-R), which refines DepthSeg using SAM; (ii) SAM-Segmentation (SAM-S), which integrates SAM with a pre-trained semantic segmentation model to improve cluster separation; and (iii) AutoSAM-Dino (ASD), which eliminates the need for manual labeling and transfer learning through the combined use of GroundingDino and SAM. The main contribution of our research is not just to improve grape bunch segmentation but also to demonstrate how zero-shot models can be effectively used to save significant time and resources that would otherwise be spent on dataset labeling and training. This is a considerable improvement over traditional methods since it enables the rapid deployment of accurate grape detection systems with little upfront work. Our experimental results show that zero-shot models can produce outcomes that are comparable to or better than existing approaches, indicating that they have the potential for practical applications in precision viticulture. The object counting and pixel-level segmentation accuracy are analyzed in comparison to a manually labeled ground truth. To this end, a dataset comprising RGB images and corresponding depth maps acquired by a farmer robot in a commercial vineyard in Southern Italy are used. The dataset has been made publicly available on GitHub at https://github.com/ispstiima/ECSDVineyardDataset.

The remainder of the paper is structured as follows. Section 2 describes the materials and methods, detailing datasets, and the proposed instance segmentation techniques. It also describes the metrics used for performance evaluation, such as mean Average Precision (mAP), Intersection over Union (IoU), and precision–recall analysis. Section 3 and 4 discuss the experimental results, providing a critical assessment of the techniques against the manually labeled ground truth. The paper concludes with a discussion of the findings, their implications for the field, and possible future research directions.

2. Materials and methods

In this section, first, the acquisition system, using a custom-built ground vehicle equipped with an RGB-D sensor is described, along with the experimental setting and the ground truth generation. Then, innovative AI models for image segmentation and grape bunch detection are presented. Finally, the performance metrics used to assess their effectiveness are introduced.

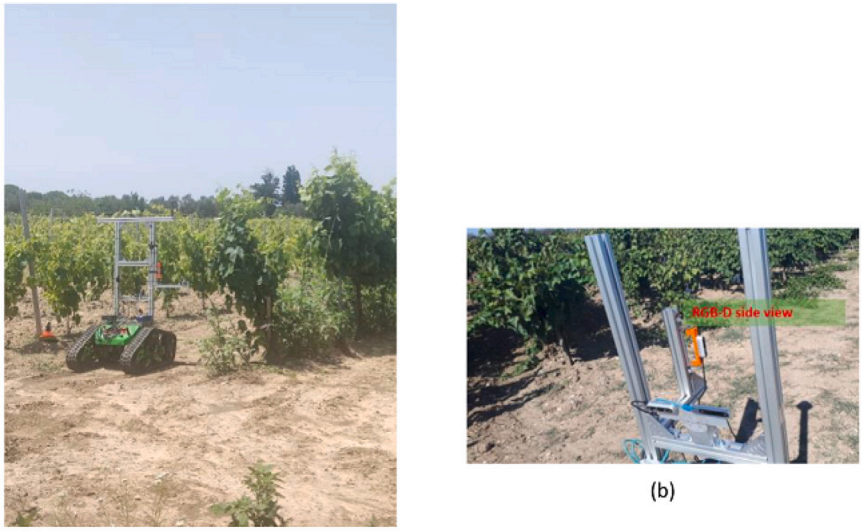

Fig. 1. The Polibot equipped with an Intel RealSense D435 imaging system: (a) navigating through vineyard rows, and (b) the camera mounted on the Polibot showing the data acquisition setup.

2.1. Data acquisition

The data for this study was collected using the Polibot, a custom-built research ground vehicle designed for high mobility over challenging terrains (Grazioso et al., 2023). The Polibot features an articulated suspension system capable of handling heavy loads, isolating vibrations, and navigating rough terrain, much like a multi-legged insect. The control and acquisition systems are implemented under the Robot Operating System (ROS), providing a flexible and robust framework for operation. The robot is equipped with an RGB-D camera, the Intel RealSense D435 imaging system, which is used for the collection of visual and depth data. Data collection was conducted in a commercial vineyard with Negroamaro red wine grape variety, located in the province of Brindisi (Italy).

The datasets were acquired by guiding the robot throughout three parcels within the vineyard plot. Each parcel spanned two rows and comprised ten plants, with five plants per row. The robot vehicle was guided along different rows, see Fig. 1(a), while the camera, mounted 1 meter above the ground, captured lateral views from an approximate distance of 1.5 m, see Fig. 1(b). The images were recorded at a resolution of 1280×720 and a frame rate of 6 Hz. This frame rate was selected as an optimal balance between computational load and grape coverage, considering that the farmer robot travels at an average speed of about 0.5 m/s. We conducted data acquisition over multiple seasons, capturing images under a wide range of weather conditions, including sunny, partially cloudy, and cloudy days. The vineyard environment was completely uncontrolled, with varying levels of sunlight exposure and natural obstructions due to leaves and branches. Such a variety of conditions allowed us to assess the robustness and scalability of the proposed methods under real-world scenarios. Among the plants in the vineyard rows, fifteen were selected for this study.

2.2. Data labeling

For the establishment of a replicable ground truth, a manual labeling process was undertaken on a set of 30 images depicting vine plants, encompassing a total of 238 instances. Each grape bunch within the images was individually labeled to ensure instance-level precision. The labeling was performed using the annotation tool LabelMe (Anon, 2023), enabling the assignment of a unique identifier to each grape cluster. This dataset serves as a benchmark for evaluating the performance of the segmentation algorithms developed in this study.

2.3. Instance segmentation methods

Three approaches are developed aimed at enhancing instance segmentation of grape bunches while minimizing the need for extensive data labeling and model training, i.e.:

1. **SAM-Refine (SAM-R):** This approach utilizes SAM to refine the clustering results obtained from a previously proposed depth-based clustering algorithm (Devanna et al., 2023). While this approach begins with a pre-trained semantic segmentation model requiring some labeled data and training, it eliminates the need for additional instance-level labeling and training. In SAM-R, depth information from an RGB-D camera is used to calculate the centroid positions of grape bunch clusters. This is achieved through a depth-based clustering algorithm that identifies clusters by detecting significant changes in depth values, effectively separating adjacent grape bunches based on depth discontinuities. These centroid positions are then used as point prompts for SAM, which generates precise instance masks for each grape bunch. By integrating SAM's zero-shot capabilities with depth data, we enhance the accuracy of instance segmentation without the need for instance-level labeling or training.
2. **SAM-Segmentation (SAM-S):** This method focuses on the segmentation masks generated by the semantic segmentation network. It employs SAM's ability to perform instance segmentation directly from semantic masks without utilizing depth data. Similar to SAM-R, it starts with a pre-trained semantic segmentation model but eliminates the need for depth information, reducing reliance on additional sensors. SAM-S reduces the effort required for instance-level data labeling and model training by utilizing SAM to separate individual grape bunches from the semantic segmentation output, thus simplifying the process compared to traditional methods.
3. **AutoSAM-Dino (ASD):** This method fully leverages zero-shot models by combining GroundingDINO to autonomously detect grape bunches without any pre-existing data or prior training, and subsequently utilizing SAM to generate precise instance masks from the detected bounding boxes. ASD completely eliminates the need for manual labeling and transfer learning for both semantic and instance segmentation, and does not require depth information, highlighting the potential of fully zero-shot models to minimize the overall effort in data preparation, model development, and reliance on additional sensor data.

Fig. 2. Comparison of vineyard imagery: (a) Original RGB image, and (b) Clustering result using the depth-based algorithm.

Table 1
Comparison of instance segmentation methods in terms of data requirements and effort.

Method	Semantic Segmentation Training Required	Instance-Level Labeling Required	Depth Data Required	Manual Labeling Effort	Model Training Effort
DepthSeg	Yes	No	Yes	Medium	High
SAM-R	Yes	No	Yes	Medium	Medium
SAM-S	Yes	No	No	Medium	Medium
ASD	No	No	No	Low	Low

These methods are compared against our original technique, referred to as DepthSeg, which combines a semantic segmentation network with a computer vision algorithm that separates instances based on depth data from an RGB-D sensor.

To better illustrate the differences among the methods in terms of data requirements and effort, a comparison is provided in Table 1.

In the following, each approach will be described in detail. The sample image of Fig. 2(a) will be used as a running sample case throughout the section for comparison of the outcomes achieved by the different segmentation techniques.

2.3.1. SAM-Refine (SAM-R)

This approach employs SAM's predictive capabilities to refine instance masks generated by the originally proposed depth-based clustering algorithm named DepthSeg (Devanna et al., 2023). In Fig. 2(b) the result of DepthSeg is shown for the running test case of Fig. 2(a). DepthSeg uses a pre-trained semantic segmentation network (e.g., DeepLabV3+) refined with in-field images, to separate fruit from non-fruit regions. Successively, the depth map is employed to define each grape cluster based on depth gradient discontinuities. It can be noticed that while most grape instances are correctly identified, contours are in some cases imprecise, leading to misdetection of grape shape. SAM-R introduces a refinement process that aims at improving the result of DepthSeg, enhancing the delineation of object contours.

The processing stages of SAM-Refine for the running sample case are shown in Fig. 3. The process begins with the creation of bounding boxes around the instance masks centroids generated by DepthSeg, which are then fed as input into SAM's predictor (see Fig. 3(a)). Utilizing a U-Net architecture, SAM's predictor first produces a coarse mask (see Fig. 3(b)), which is then refined through a convolutional network to produce the final object mask for each grape bunch (see Fig. 3(c)). It can be noticed that the bunch shape is improved with respect to DepthSeg results. This may be beneficial for applications such as shape reconstruction or biomass estimation.

2.3.2. SAM-Segmentation (SAM-S)

The second approach exploits the capability of SAM's mask generator of automatically segmenting the visual scene into clusters. In order to detect only clusters pertaining to the grape class, grape pixels are first isolated based on semantic segmentation (see Fig. 4(a)). Then, the SAM mask generator is applied to identify single grape instances (see Fig. 4(b)).

Let us consider that, while some grape bunches may not be detected, this method offers significant advantages by eliminating the need for depth data and centroid calculations. This simplification reduces the data acquisition burden and computational requirements, making it more feasible for deployment in various agricultural settings. The trade-off between detection accuracy and resource efficiency is an important consideration, and SAM-S provides a balanced approach that aligns with our objective of minimizing extensive data labeling and model training.

2.3.3. AutoSAM-Dino (ASD)

The third approach explores the synergistic application of GroundingDINO and SAM for instance segmentation. Upon receiving the keyword *grape*, GroundingDINO autonomously identifies and creates bounding boxes for grape bunches (see Fig. 5(a)). These are then fed into SAM's predictor, which generates instance masks for each identified bunch, as shown in Fig. 5(b). This strategy capitalizes on the

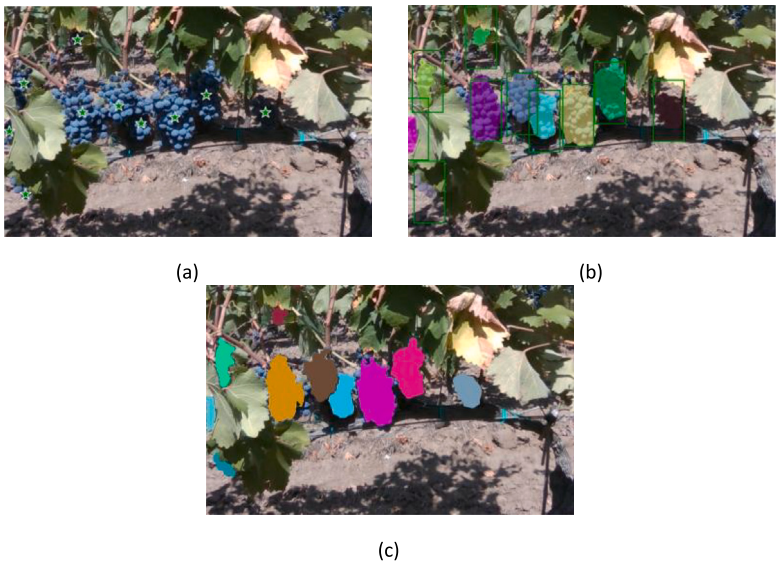

Fig. 3. SAM-R workflow: (a) Bounding box seeds from the depth-based clustering algorithm, (b) Coarse masks generated by SAM's predictor, (c) Overlay of the refined masks on the original RGB image.

Fig. 4. SAM-S workflow: (a) masked RGB image used as input to SAM, (b) segmented image with SAM's outcome of grape clusters overlaid on the original RGB.

advanced capabilities of both models to expedite the object detection and the instance segmentation process, significantly reducing the time and resources typically required for data preparation, such as data collection, labeling, and model training. While this method offers considerable time savings and operational efficiency, it also presents limitations in terms of fine-tuning and adaptability. Unlike algorithmic approaches where the user can modify parameters or intervene directly in the processing steps, the use of pre-trained models like GroundingDINO and SAM provides less flexibility. If the models do not perform as anticipated, the user has limited options for adjustments, as the internal workings of these models are not as accessible for modifications. This lack of control can be a trade-off for the convenience and speed of using such advanced, pre-trained systems.

2.4. Metrics for performance evaluation

A comprehensive evaluation framework is employed to assess the performance of the proposed models against the DepthSeg algorithm, serving as a benchmark. The framework encompasses quantitative metrics aimed at evaluating the efficacy of the techniques in enhancing results, with respect to both their object detection capability and pixelwise instance segmentation accuracy.

Initially, the assessment considers how well the approach identifies objects of interest, classifying each instance detected by the model as a True Positive (TP) or False Positive (FP) irrespective of the ground truth mask. This binary appraisal necessitates a rigorous method to determine TP instances, which involves the association of predictions to their corresponding ground truth targets.

This is a challenging task that necessitates a clear definition and identification process. Inspired by methodologies available in the literature (Padilla et al., 2020), the adopted approach involves calculating the Intersection over Union (IoU) for each predicted instance against every ground truth instance. Hence, for an image X with n predicted instances and m ground truth instances, this calculation results in an $m \times n$ matrix of IoU values, offering a numerical representation of the overlap between predicted and actual instances, as can be seen in Fig. 6. A matching algorithm is devised to associate a predicted instance with its actual ground truth counterpart. This algorithm addresses ambiguous and special cases, allowing the selection of an overlap threshold as the primary discriminator for determining TP from FP instances. In cases where multiple predictions correspond to a single ground truth instance, the prediction with the highest overlap is deemed TP, while others are marked as FP. The threshold for the minimum IoU necessary for an instance to be considered TP is set at approximately one-third of the predicted instance's surface area (IoU > 0.3).

Based on TPs and FPs values, precision and recall metrics are cumulatively computed across all instances within the test dataset. The average precision for each instance is calculated and the precision–recall curve is constructed. The mean Average Precision (mAP) is then determined using the 11-point interpolation technique.

Subsequently, the instance segmentation performance on TPs at pixel level is evaluated using the following metrics:

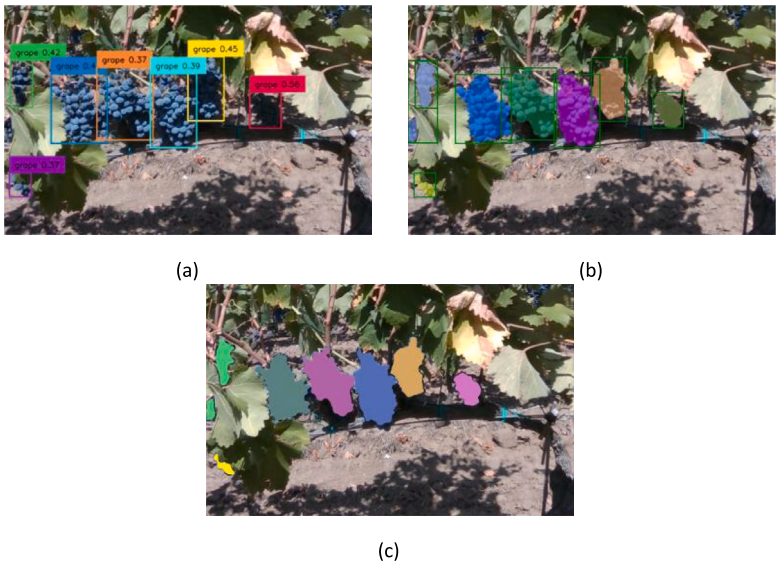

Fig. 5. ASD workflow: (a) Initial detection by GroundingDINO, utilizing the keyword *grape* to autonomously create bounding boxes around grape clusters, (b) Integration of SAM's predictive capabilities, refining the bounding boxes from GroundingDINO into precise instance masks for each grape cluster, (c) Final segmentation results overlaying the refined instance masks on the original RGB image.

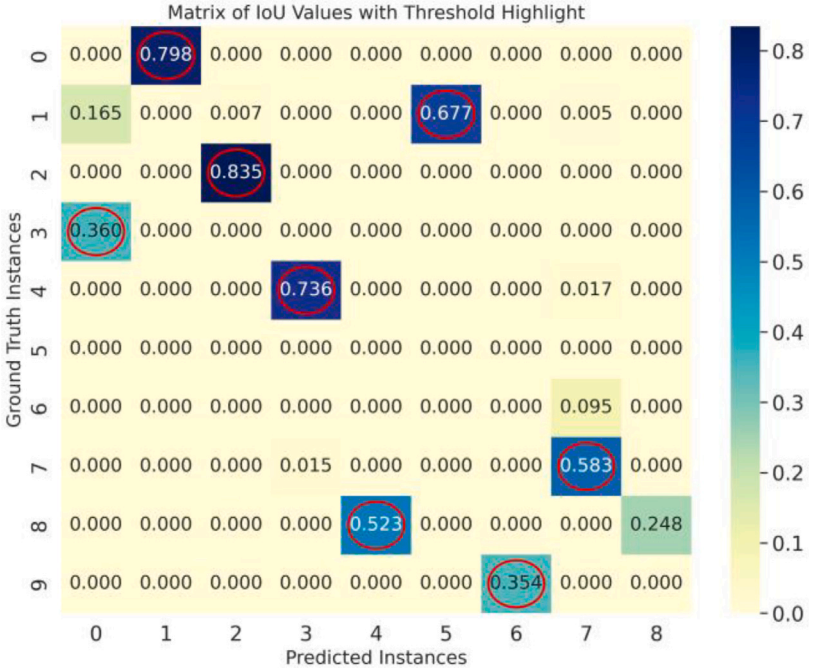

Fig. 6. Intersection over Union (IoU) matrix visualization for Fig. 2(b), a DepthSeg instance segmentation result, consisting of 9 predicted instances (columns) and 10 ground truth instances (rows). Each cell's color intensity represents the IoU score, with colder hues indicating greater overlap between predicted and actual instances. Red circles highlight IoU values that exceed the 0.3 threshold, designating them as potential True Positives (TP) in the object detection evaluation process. This matrix effectively encapsulates the model's performance in identifying and delineating objects within the image, facilitating a quantitative analysis of detection accuracy.

- **IoU:** reflects the exactness of the segmentation, with a higher IoU indicating a more accurate shape.
- **Precision:** assesses the fraction of pixels correctly identified as part of the instance against the total predicted pixels.
- **Recall:** evaluates the ability of the model to include all actual instance pixels in its prediction.

This pixel-level analysis reveals the performance of each approach in segmenting every instance. By calculating these metrics for each instance and then averaging them, the mean IoU, Precision and Recall for all instances are also calculated.

In order to visually represent the performance of the different methods, a representation such as the one shown in Fig. 7 is adopted. Specifically, for object detection, a bounding box visualization is used following a color-coded scheme (see left column of Fig. 7), i.e.:

- Yellow boxes denote the ground truth instances.
- Green boxes represent the predicted instances correctly identified as True Positives (TP).
- Red boxes indicate the False Positives (FP), where the predicted instances did not correspond to the ground truth.

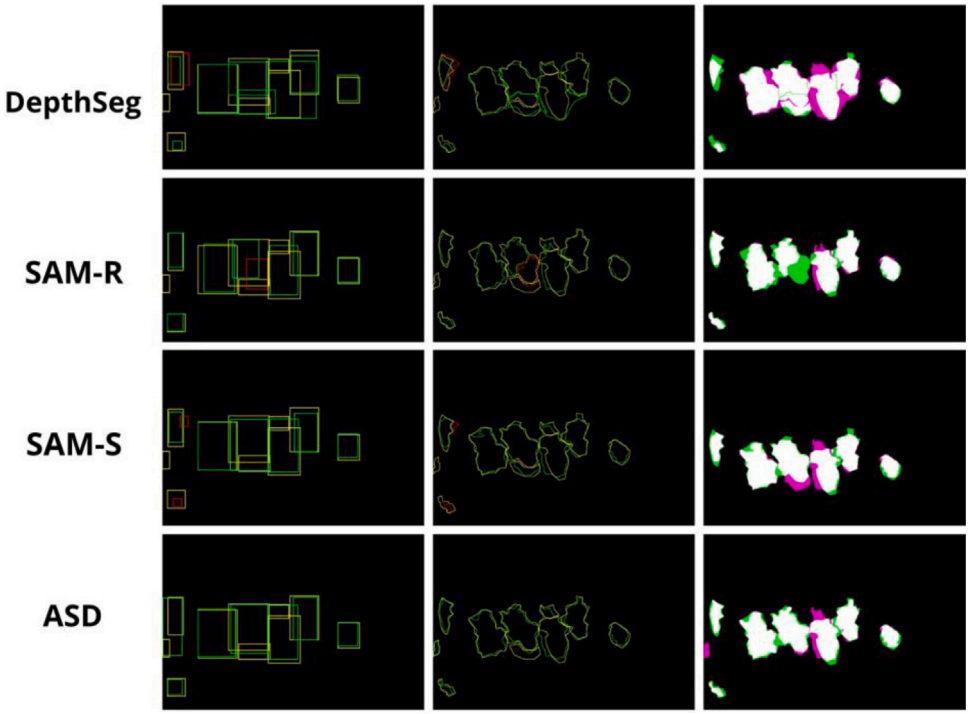

Fig. 7. In the columns, from left to right: i) Bounding box comparison between the ground truth (yellow) and the SAM-Refine model predictions, with true positives (TPs) in green and false positives (FPs) in red. ii) Edge-based representation of grape cluster boundaries, providing a clear view of how the SAM-Refine model delineates the shapes of TPs against the ground truth. iii) Segmentation evaluation focusing on TPs. White pixels represent the correctly segmented TPs, magenta pixels indicate false positives (FPs), and green pixels show the regions of false negatives (FNs).

Since dense clustering or overlapping boxes could compromise the clarity of this representation, an alternative graphical representation is devised, which traces the contours of each instance (see central column of Fig. 7). Note that this is not intended to demonstrate the model's segmentation capability.

In addition, for visualization of instance segmentation at pixel level, segmentation result overlap is shown as follows (see right columns of Fig. 7):

- Correctly identified pixels are displayed in white, signifying True Positives (TP).
- Pixels erroneously included in the predicted mask are highlighted in magenta, representing False Positives (FP).
- Pixels incorrectly excluded from the predicted mask are marked in green, indicating False Negatives (FN).

3. Experimental results

This section provides a quantitative evaluation of the proposed instance segmentation techniques based on the metrics presented in Section 2.4. In Fig. 8 the precision–recall curves are shown for each method. From them, the mAP has been evaluated, and the results are reported in the bar chart of Fig. 9.

DepthSeg establishes a baseline, achieving a mAP of 0.590. SAM-Refine (SAM-R) demonstrates a marginal improvement in mAP to 0.593, a +0.5% increase (see Fig. 10), suggesting that refining instance masks with SAM can slightly enhance object detection accuracy compared to the depth-based clustering algorithm. In contrast, SAM-Segmentation (SAM-S) shows a decrease in mAP to 0.515, leading to a −12.7% reduction, and AutoSAM-DINO (ASD) decreases further to 0.438, marking a −25.8% change.

At pixel segmentation level, all three approaches show comparable performance in terms of IoU, with SAM-R and SAM-S marginally surpassing DepthSeg. This suggests a similar ability to achieve overlap between the predicted masks and the ground truth. As regards precision and recall performance, SAM-R attains the highest precision, underscoring its effectiveness in correctly identifying instance pixels, while ASD exhibits a significant increase in recall, indicating a more comprehensive inclusion of actual instance pixels in its predictions. Visual insights about segmentation results are depicted in bar charts in Figs. 11 and 12. In summary, DepthSeg, serving as the baseline, demonstrated a good performance with a mAP of 0.590, suggesting that the depth-based approach retains its relevance in object detection. However, the marginally improved mAP of 0.593 achieved by SAM-Refine (SAM-R) indicates that even subtle algorithmic refinements can enhance object detection accuracy. The slight reduction in mAP observed in SAM-Segmentation (SAM-S) to 0.515 and further to 0.438 in AutoSAM-Dino (ASD) raises some considerations. While at first glance, this may appear as a regression in model performance, the underlying reasons are multifaceted. The SAM-S model, which does not use depth data, indicates that less information does not necessarily lead to lower performance, given that the segmentation accuracy in terms of IoU has seen an improvement. This suggests a trade-off between the number of instances detected and the quality of segmentation. ASD's performance is particularly noteworthy. Despite a lower mAP, it demonstrates enhanced accuracy in segmentation, which is evidence of the model's zero-shot semantic segmentation capabilities without requiring a training stage. However, the precision–recall trade-off inherent in this model suggests a more complex narrative. While ASD can maintain high precision across a range of instances, the steep decline beyond a certain recall threshold reflects its limitations in clustering performance, potentially leading to many true instances going undetected.

4. Discussion

Here, results obtained at both object detection and pixel segmentation levels are discussed in more detail, with reference to additional

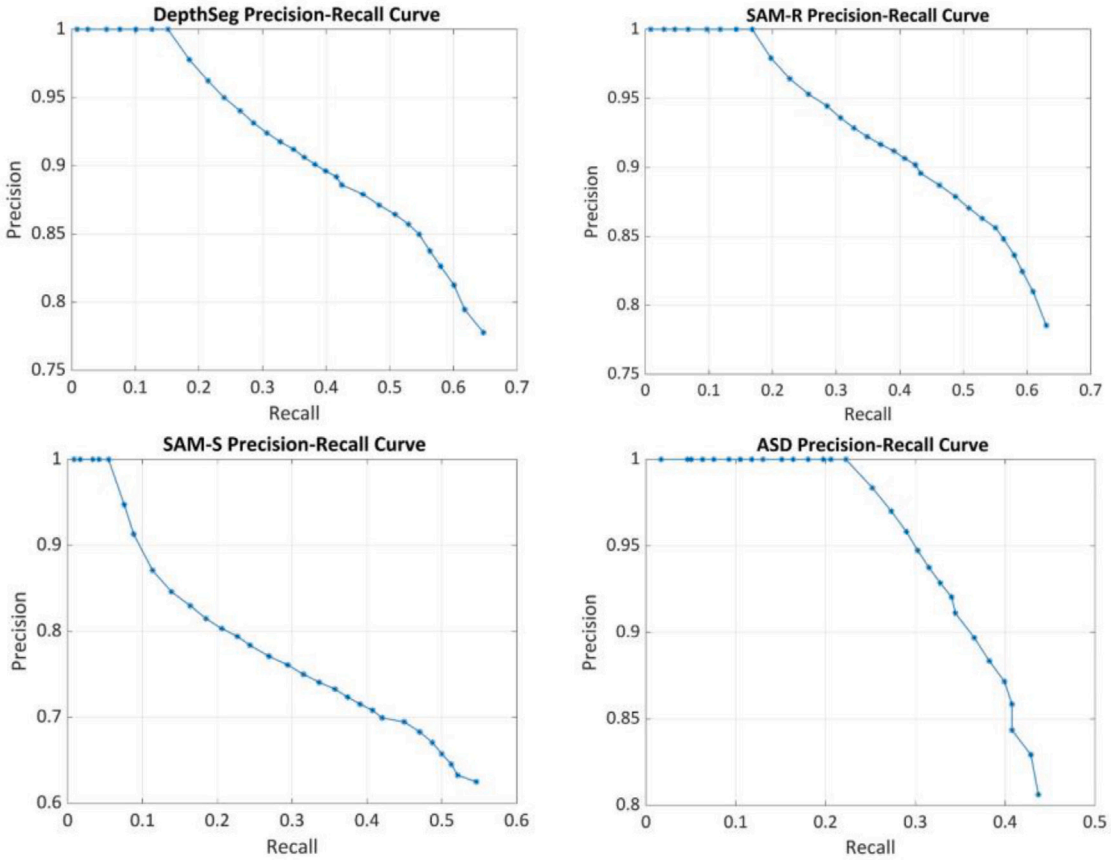

Fig. 8. Precision–Recall Curves illustrating the performance of each segmentation model: DepthSeg, SAM-R, SAM-S, and ASD—used, in the evaluation of instance segmentation methods. Each curve depicts the relationship between accumulated precision (y-axis) and accumulated recall (x-axis), contributing to the computation of the model's mean Average Precision (mAP). The curves encapsulate the comprehensive detection capabilities of each model, with the area under each curve being indicative of the overall precision and recall balance. The quantified mAP from these curves serves as a benchmark for comparison across the models.

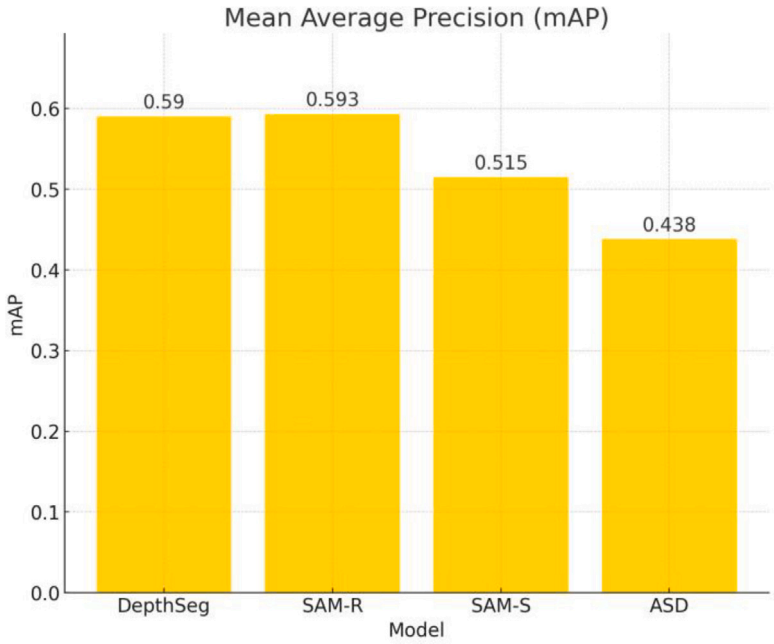

Fig. 9. Bar chart illustrating the mean average precision (mAP) for each segmentation method. DepthSeg serves as the mAP benchmark, while SAM-Refine (SAM-R) shows a marginal improvement. SAM-Segmentation (SAM-S) and AutoSAM-DINO (ASD) exhibit reductions in mAP.

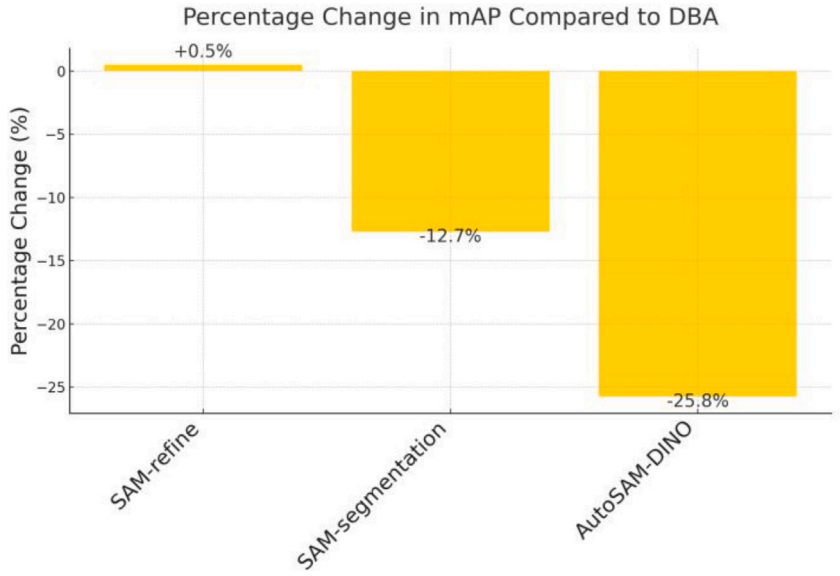

Fig. 10. Percentage change in mean average precision (mAP) relative to DepthSeg. Positive values indicate an improvement over the benchmark, with SAM-Refine (SAM-R) registering a slight increase. Negative values denote a decrease, as seen with SAM-Segmentation (SAM-S) and AutoSAM-DINO (ASD).

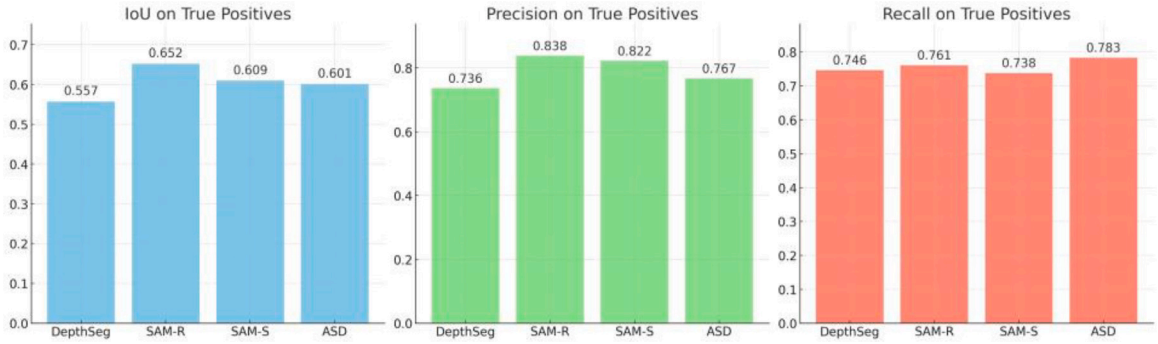

Fig. 11. Comparative analysis of Intersection over Union (IoU), precision, and recall metrics for true positive instances across the segmentation approaches. Each method's performance is evaluated, revealing their relative strengths and weaknesses in instance segmentation accuracy.

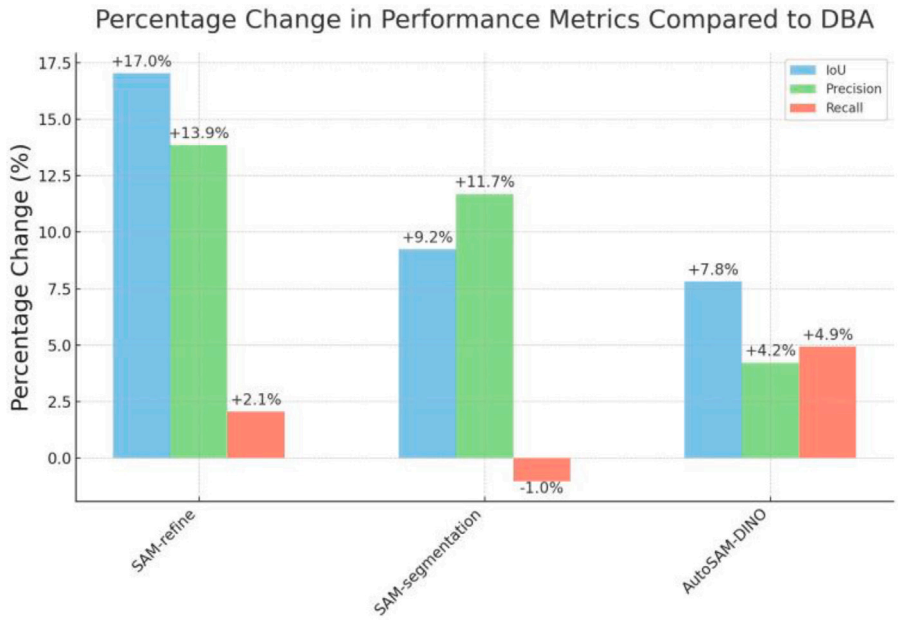

Fig. 12. Graph showing the percentage change in performance metrics, including IoU, precision, and recall, compared to DepthSeg. The variations are indicative of each method's improvement or decline in segmenting instances accurately.

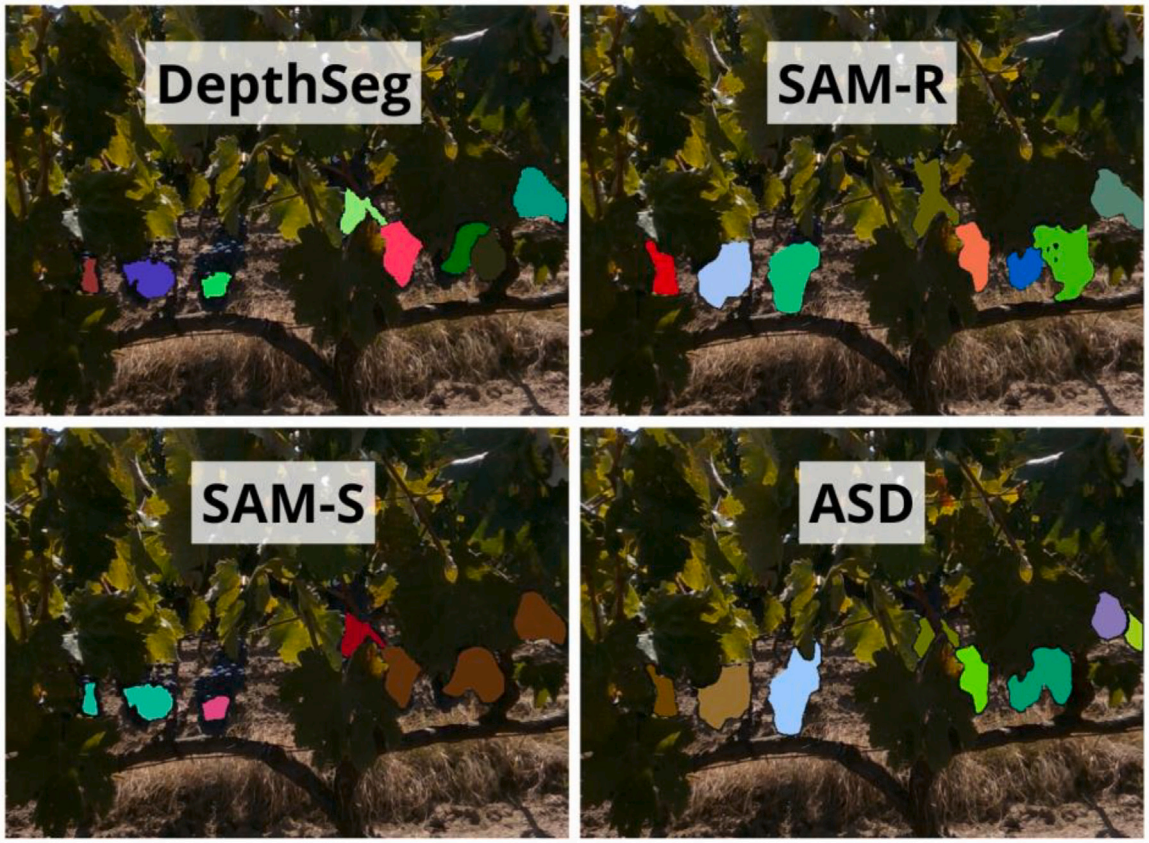

Fig. 13. Comparative Visualization Under Poor Lighting. These images illustrate the performance of the four segmentation models, where DepthSeg and SAM-R utilize depth data to discern clusters, and ASD demonstrates enhanced segmentation despite challenging lighting conditions.

sample cases obtained under different conditions. Specifically, the influence of environmental lighting, quality of depth data, and accuracy of semantic segmentation are analyzed.

4.1. Influence of lighting on model performance

In precision agriculture, the role of lighting conditions is paramount, as it directly impacts the quality of visual data and, consequently, the performance of segmentation models. DepthSeg and SAM-Refine (SAM-R) use depth data, which becomes particularly beneficial under poor lighting. This advantage is evident in their ability to discern clusters that are not distinguishable in standard RGB imagery due to low illumination levels. It suggests that depth data can serve as a critical supplement to visual information when light is deficient. An example is shown in Fig. 13, where SAM-S and ASD fail to separate some grape instances due to poor RGB information.

In addition, SAM-S suffers when initial semantic segmentation quality is compromised, which can be further worsened by inadequate lighting, while AutoSAM-Dino (ASD) demonstrates a considerable improvement in segmentation under these challenging conditions. Despite the less effective isolation of individual instances, ASD's segmentation quality is notable, signifying the potential of advanced zero-shot learning models to adapt to variable lighting in agricultural environments.

The performance of the approaches for this test case are visually represented in Fig. 14.

For DepthSeg, the yellow bounding boxes of the ground truth overlap with the model's green true positives, indicating a moderate success rate in accurately identifying clusters. However, there is a false positive instance that did not pass the IoU threshold and has been considered a false positive due to the low overlap with its respective ground truth (highlighting the role of initial semantic segmentation quality, which can be compromised without adequate lighting.).

SAM-Refine (SAM-R) exhibits an improvement, with green boxes more aligned to the ground truth. This alignment denotes SAM-R's enhanced ability to refine segmentation, leveraging depth information effectively even in low-light scenarios. The colored edges show a closer adherence to the true shapes of clusters, underscoring the precision of SAM-R's refinement process.

SAM-Segmentation (SAM-S), on the other hand, shows a scattering of both green true positives and red false positives. The edges, while outlining the instances, occasionally merge with adjacent clusters, reflecting the model's struggle with distinct segmentation in poor lighting.

AutoSAM-Dino (ASD) demonstrates a more consistent overlay of green true positives, suggesting its robustness in challenging lighting conditions. However, the magenta pixels of false positives within the masks and occasional large green bounding boxes enveloping multiple ground truth clusters indicate instances where ASD's zero-shot capabilities have misinterpreted adjacent clusters as a single instance due to the lack of visual clarity.

4.2. Robustness to noisy depth data

Corrupted or noisy depth data represent another condition that may decrease the performance of the methods relying on detph, i.e. DepthSeg and SAM-R. This can be seen for the sample case of Fig. 15. It can be noticed that DepthSeg leads to an over-fragmenting of the biggest clusters. This fragmentation underscores the challenges of relying heavily on depth data that may not always be accurate.

SAM-R partially mitigates the inaccuracies introduced by erroneous depth data through a refinement of the cluster shapes, thus showing a better capability to filter out noise and enhance the reliability of instance segmentation.

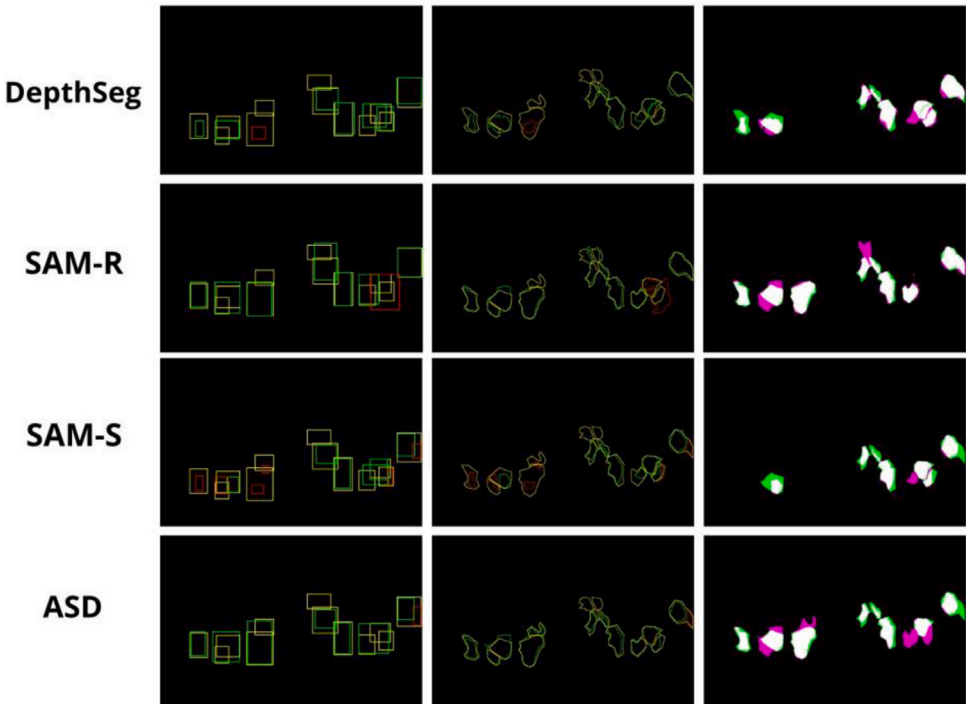

Fig. 14. Visual instance segmentation results representation, evaluated under poor lighting condition.

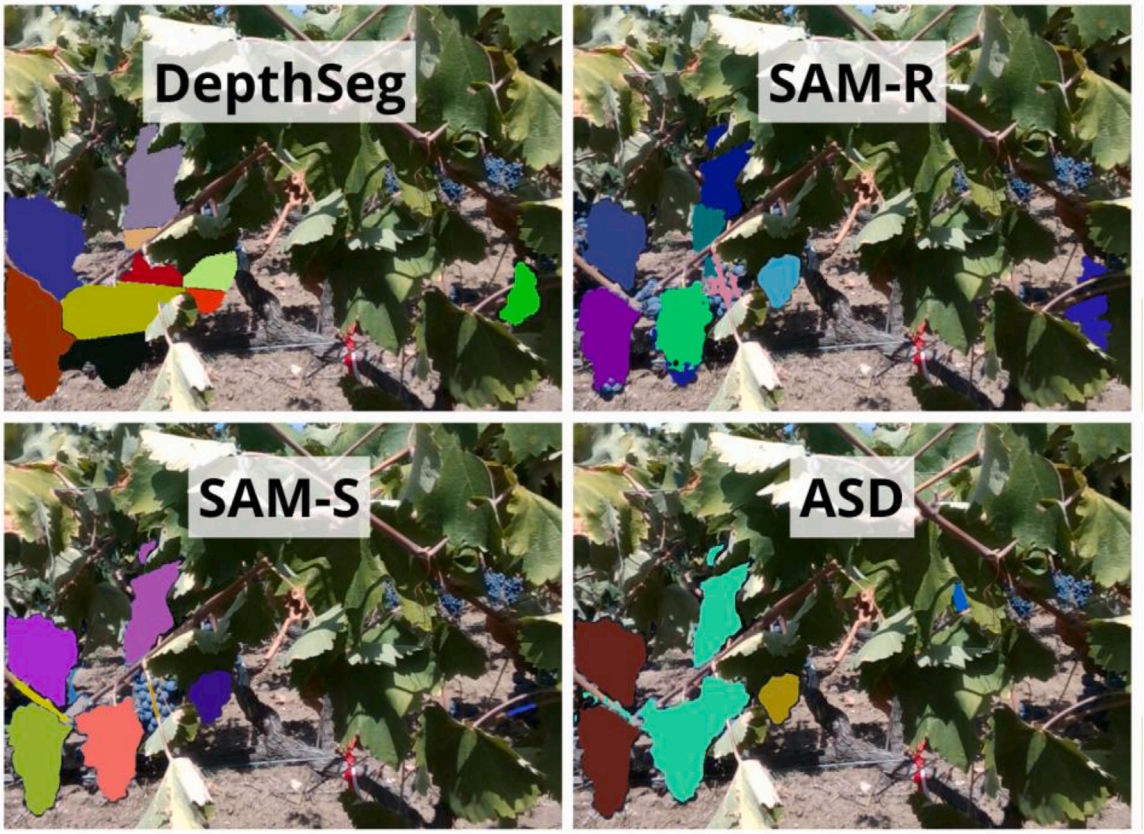

Fig. 15. Segmentation with Noisy Depth Data: Displayed here are the instances of grape clusters identified by each model, with SAM-R's refinement capabilities being notably robust against noisy depth data.

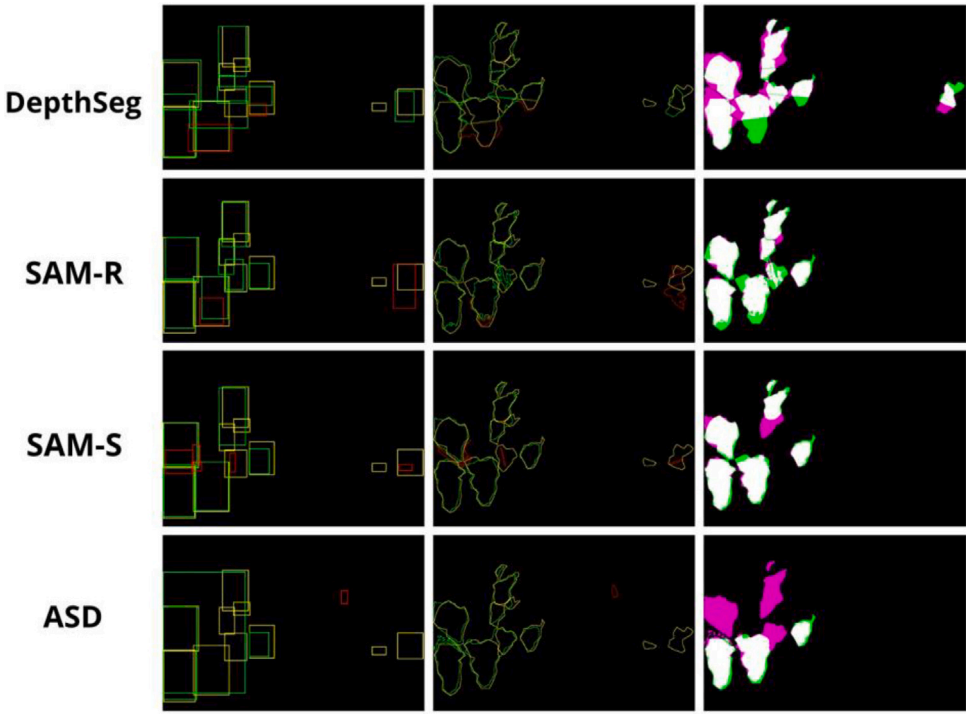

Fig. 16. Visual instance segmentation results representation, evaluated under noisy depth condition.

The visual representation of the segmentation models in the case of noisy depth data reported in Fig. 16 and provide a clear depiction of the behavior of each model.

For DepthSeg, the green bounding boxes show where the model has correctly identified clusters, aligning with the yellow ground truth. However, the presence of red false positives, especially in cases where bounding boxes are oversized or incorrectly placed, highlights the model's susceptibility to noise in the depth data. These red false positives suggest an over-segmentation tendency, where DepthSeg is interpreting noise as separate instances.

SAM-Refine (SAM-R) displays a marked improvement. Its bounding boxes are more accurately placed, and the edges are more refined, closely tracing the true shapes of the grape clusters. This precision in defining the boundaries of each cluster, even in the presence of noise, showcases SAM-R's effective noise-filtering capabilities. The improved overlap in the mask overlays further corroborates this assumption.

For this test case, the results of SAM-S and ASD, which are not affected by depth, are also shown in Figs. 15 and 16 for the sake of completeness.

4.3. Semantic segmentation quality

The dependency on initial semantic segmentation quality is a common thread across DepthSeg, SAM-R, and SAM-S. An example is shown in Fig. 17. It can be seen that DepthSeg, SAM-R, and SAM-S perform poorly due to the loss of multiple grapes in the initial segmentation phase. However, SAM-R leads to better individual grape cluster delineation, indicating its effectiveness in enhancing pixel level segmentation. ASD's independence from the DepthSeg pipeline allows it to excel beyond the limitations faced by the other models. Being independent of the initial segmentation's quality, ASD achieves the best results.

With reference to Fig. 18, it can be noticed that in the cases of DepthSeg, SAM-R, and SAM-S, the green bounding boxes that correspond to true positives are confined to the areas output by the initial segmentation. This reveals their dependency on the quality of the initial segmentation cues, with the number of true positives being limited to the instances first recognized in the semantic segmentation phase.

AutoSAM-Dino (ASD), in contrast, demonstrates its capability to transcend the limitations of the initial segmentation quality. The bounding boxes and masks for ASD are not constrained by the initial segmentation boundaries, as evidenced by the presence of true positive detections even outside the regions identified by the initial semantic segmentation. This indicates ASD's ability to independently identify and segment instances based on its zero-shot learning, unaffected by any prior segmentation errors or omissions.

4.3.1. Qualitative analysis

In this section, a qualitative analysis of the proposed approaches is performed, evaluating the required effort for training and implementation as well as the degree of transparency of the model parameters to the user.

Effort Evaluation:

The operational deployment of AI models extends beyond numerical accuracy; it encompasses the practicality of their implementation. The following evaluation metrics are defined as indicators of resource requirements of each model:

Data Requirement: This metric evaluates the extent of data preparation necessary for the successful application of the segmentation model. It is rated on a scale from 1 to 3, where:

- **1 (Low)**: No new data acquisition or labeling is required, allowing for immediate deployment.
- **2 (Medium)**: New data collection and manual labeling are necessary, indicating a moderate need for resources.
- **3 (High)**: New data collection, labeling, and the incorporation of additional data types, such as depth information, are required, reflecting significant resource investment.

Preparation Time: This metric assesses the time investment required to set all the segmentation models until they reach operational efficacy. It is also rated on a scale from 1 to 3, where:

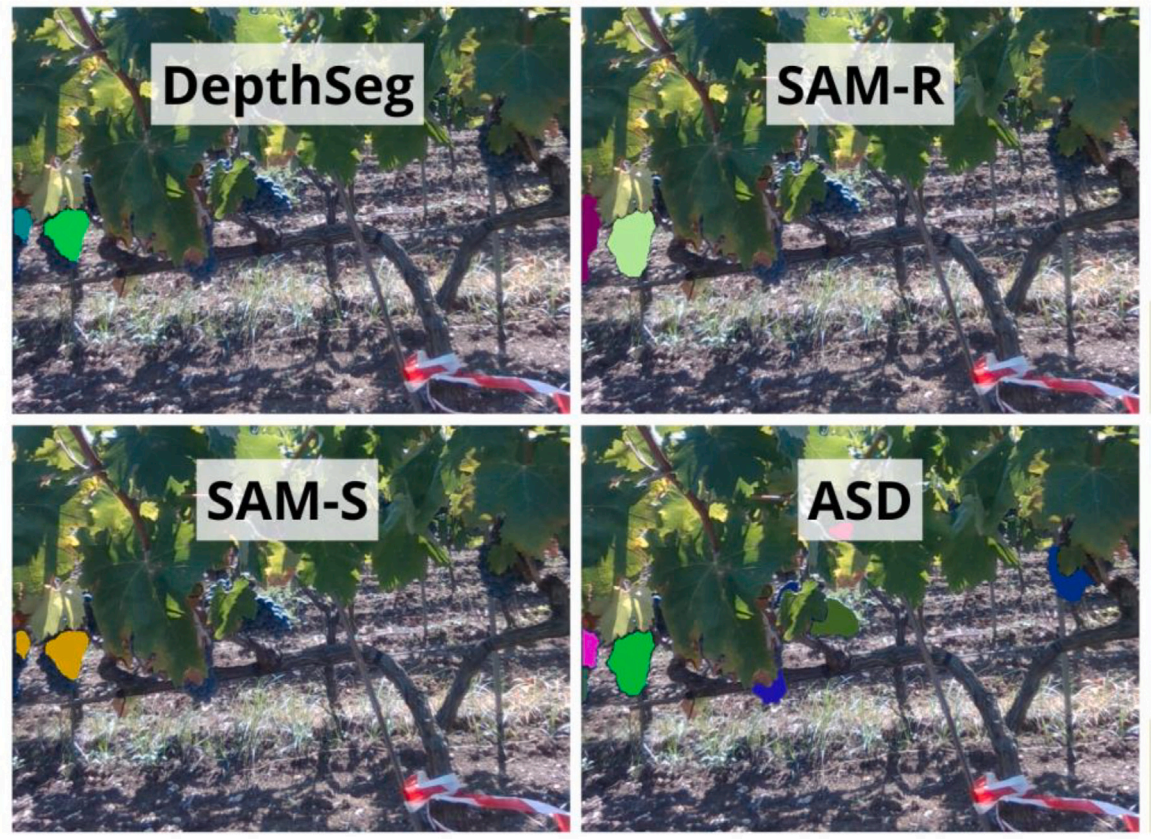

Fig. 17. Impact of Initial Segmentation Quality: The visual contrasts here highlight the dependency of DepthSeg, SAM-R, and SAM-S on the quality of initial segmentation and the superior performance of ASD in segment the scenes.

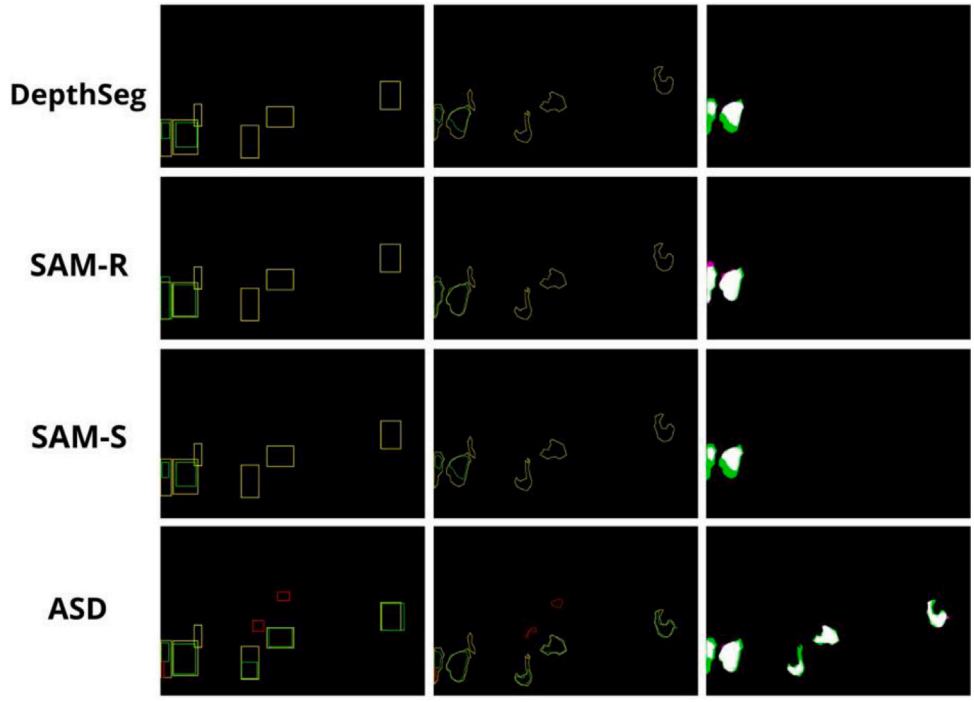

Fig. 18. Visual instance segmentation results representation, evaluated under poor semantic segmentation condition (DepthSeg, SAM-R, SAM-S).

Table 2
Summary of effort and transparency scores for each segmentation method.

Method	Effort Determinants			Effort	Transp.
	Data Req.	Train. Time	Prep. Complex.	Score	
DepthSeg	3	3	3	3	2
SAM-R	3	3	3	3	2
SAM-S	2	2	2	2	3
ASD	1	1	1	1	3

- **1 (Low)**: Models employ zero-shot detection and segmentation techniques without the need for additional training time.
- **2 (Medium)**: A consistent training time is required, including the manual labeling of ground truth training data.
- **3 (High)**: A consistent training time is required, including the manual labeling of ground truth training data and the potential development of on-purpose algorithms.

Preparation Complexity: This metric indicates the complexity of the pre-processing for segmentation deployment. It spans a scale from 1 to 3, where:

- **1 (Low)**: The model requires minimal preparatory steps, simplifying the deployment process.
- **2 (Medium)**: Moderate preparatory activities are needed, which may involve some level of data processing or parameter tuning.
- **3 (High)**: Intensive preparation is imperative, which could entail extensive manual labeling or algorithmic customization.

The final effort score for each method is computed by averaging the individual scores assigned to these criteria, thus yielding an aggregate indicator of the resource requirement.

Transparency Evaluation:

This score is related to the transparency of the models, i.e., the degree to which a user can intervene in the model's parameters and decision-making process. It is measured on a scale from 1 to 3, where:

- **1 (High)**: The model operates with full transparency, offering the user complete access to the algorithms and the ability to make modifications as needed.
- **2 (Medium)**: There exists a semi-transparent operation with a moderate level of understanding and control.
- **3 (Low)**: The model functions are "black boxe" with internal processes being largely inaccessible or not readily interpretable by the user.

In Table 2 an overview of the effort and transparency scores for each segmentation method examined in the study is reported. Incorporating these operational aspects into the discussion enriches our understanding of the models' real-world applicability. It provides information that goes beyond quantitative metrics analysis, highlighting the importance of practical aspects, which are critical in precision farming contexts.

The effort and transparency scores, when coupled with the quantitative metrics, provide an overall evaluation of each method, as can be seen in Fig. 19. DepthSeg and SAM-R, despite their high resource requirements, offer substantial control and adaptability, which could be useful in complex agricultural scenarios. SAM-S, with its moderate resource requirements, represents a middle ground. ASD emerges as the most operationally efficient model, suggesting its potential as a quick-deployment tool in environments where speed and efficiency are prioritized over user control.

In conclusion, this integrated view allows for informed decision-making when selecting and deploying segmentation models. It underscores the need for models that not only perform well in terms of numerical accuracy but also adhere to the real operational conditions in precision agriculture settings.

5. Conclusion

In this work, three novel AI-based approaches for in-field grape instance segmentation were proposed, referred to as SAM-Refine (SAM-R), SAM-Segmentation (SAM-S), and AutoSAM-Dino (ASD), and compared with a previously proposed method named DepthSeg.

All models integrate the latest zero-shot learning techniques to boost the detection and segmentation of grape bunches using images provided by a farmer robot's on-board RGB-D camera. The ultimate goal is to provide information for agricultural tasks such as growth monitoring and yield estimation.

The performance analysis of the different approaches encompassed several aspects, concerning their efficacy and practical impact on enhancing object detection and semantic segmentation.

Experimental results obtained for a dataset acquired in a commercial farm in southern Italy uncovered significant variations in their performance within the specific realm of precision agriculture.

DepthSeg, the benchmark model developed using conventional deep learning segmentation networks and computer vision techniques, in conjunction with depth data for clustering, demonstrated good object detection capabilities. This is crucial for tasks such as fruit counting, thereby facilitating applications related to yield estimation and evaluation, with a mean Average Precision (mAP) of 0.590. However, its instance segmentation ability is not optimal, presenting challenges in applications where accurate fruit shape information is essential for phenotyping.

In comparison with DepthSeg, SAM-R, integrating the depth-based segmentation technique with the Meta's SAM model, showed slight improvements in counting and a substantial enhancement in segmentation. Conversely, SAM-S and ASD, despite their innovative approaches and commendable segmentation values, exhibited a decrease in mAP, indicating a compromise between object detection accuracy and segmentation quality.

The influence of external factors, such as lighting conditions and data quality, on model performance was also investigated. Depth data-based models, like DepthSeg and SAM-R, demonstrated good performance under suboptimal lighting conditions, benefiting from the additional depth information. This aspect is particularly advantageous in precision agriculture, where environmental variability is common. Conversely, models relying exclusively on visual data, such as SAM-S and ASD, encountered challenges in similar conditions. However, if the depth data is compromised or excessively noisy, DepthSeg and SAM-R are significantly affected, leading to errors in cluster detection.

It is noteworthy that the approaches of DepthSeg, SAM-R, and SAM-S share a common dependency on the quality of prior semantic segmentation performed on RGB data using deep learning networks. Hence, they are networks retrained through transfer learning on specific datasets, which might lack substantial generalization capabilities under complex conditions. This is in contrast to the strength of SAM, and thus the ASD approach, which does not rely on prior semantic segmentation networks but only on SAM and GroundingDino models.

Furthermore, the operational aspects of deploying these models, including effort and transparency evaluations, were examined. ASD emerged as a model with low operational effort but low transparency, indicating its suitability for rapid deployment in scenarios where efficiency is prioritized. In contrast, DepthSeg and SAM-R, while demanding higher operational effort, offered greater control and adaptability,

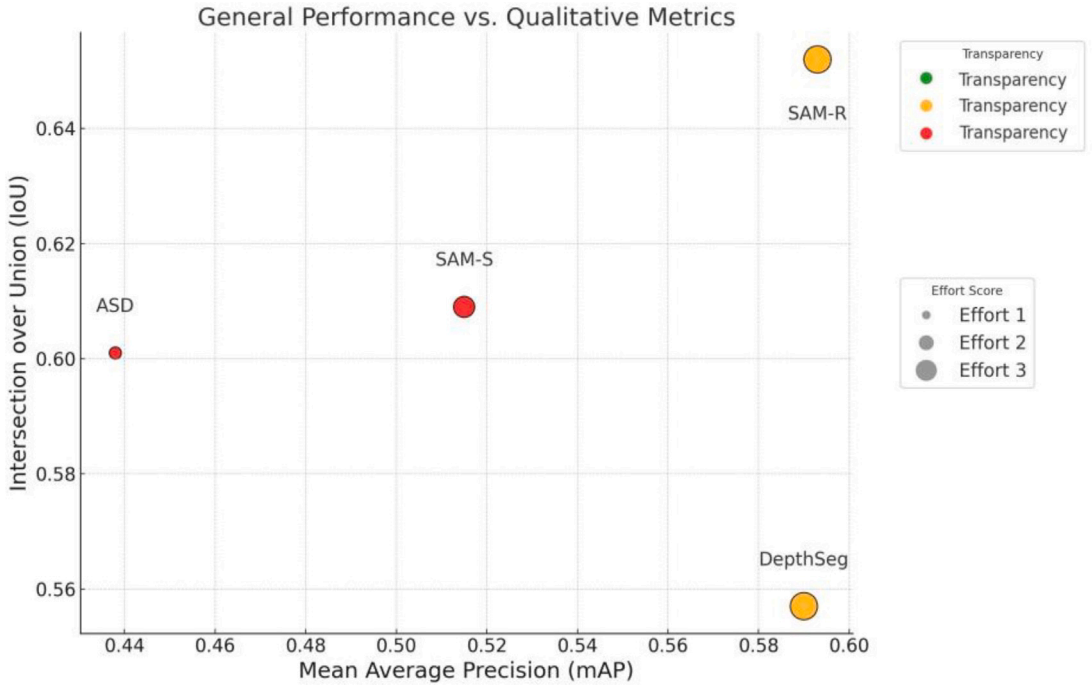

Fig. 19. Summary of effort and transparency scores for each segmentation method. The table quantifies the data requirement, training time, and preparation complexity, with a resulting effort score and transparency level. Lower effort scores denote less resource investment, and higher transparency scores indicate a more user-friendly and controllable process.

which could be crucial in complex agricultural settings. The consistent performance of our models across diverse conditions demonstrates their potential scalability to larger areas and different times or seasons. The elimination of extensive data labeling and model training requirements makes our approach highly applicable for operational uses in precision agriculture. By effectively handling the variability inherent in uncontrolled environments, our methods can be integrated into existing agricultural workflows, facilitating tasks such as yield estimation, crop monitoring, and targeted interventions.

In conclusion, this study has conducted a comprehensive comparative analysis of various AI-based segmentation methods, shedding light on their applicability, efficiency, and practicality in real-world precision agriculture scenarios. The results underscore the potential of optimizing different AI models to address specific agricultural sector challenges, thereby enhancing crop management, yield optimization, and contributing to environmental sustainability.

CRediT authorship contribution statement

Rosa Pia Devanna: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Giulio Reina:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Funding acquisition, Formal analysis, Conceptualization. **Fernando Auat Cheein:** Writing – review & editing, Writing – original draft, Supervision, Formal analysis, Conceptualization. **Annalisa Milella:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was partially funded by the projects: AgRibot - Harnessing Robotics, XR/AR, and 5G for a New Era of Safe, Sustainable, and Smart Agriculture, European Union's Horizon Europe research and innovation programme under grant agreement (No. 101183158); giving Smell sense To Agricultural Robotics (STAR), ERA-NET COFUND ICT AGRI-FOOD (Grant No. 45207); CNR DIITET project DIT.AD022.207, STRIVE-le Scienze per le TRansizioni Industriale, Verde ed Energetica (FOE 2022), sub task activity Agro-Sensing2. The authors are grateful to the agricultural farm Cantina San Donaci (BR), Italy, for hosting experimental tests.

Data availability

Data will be made available on request.

References

Anon, 2023. LabelMe. http://aiweb.techfak.uni-bielefeld.de/content/bworld-robot-control-software/. [Online; accessed 11-December-2023].

Boateng, E.Y., Otoo, J., Abaye, D.A., 2020. Basic tenets of classification algorithms K-nearest-neighbor, support vector machine, random forest and neural network: A review. J. Data Anal. Inf. Process. 8 (4), 341–357.

Casado-García, A., Heras, J., Milella, A., Marani, R., 2022. Semi-supervised deep learning and low-cost cameras for the semantic segmentation of natural images in viticulture. Precis. Agric. 1–26.

Cheng, Y., Li, L., Xu, Y., Li, X., Yang, Z., Wang, W., Yang, Y., 2023. Segment and track anything. ArXiv preprint arXiv:2305.06558.

Ciarfuglia, T.A., Motoi, I.M., Saraceni, L., Fawakherji, M., Sanfeliu, A., Nardi, D., 2023. Weakly and semi-supervised detection, segmentation and tracking of table grapes with limited and noisy data. Comput. Electron. Agric. 205, 107624.

Devanna, R.P., Milella, A., Marani, R., Garofalo, S.P., Vivaldi, G.A., Pascuzzi, S., Galati, R., Reina, G., 2022. In-field automatic identification of pomegranates using a farmer robot. Sensors 22 (15).

Devanna, R.P., Reina, G., Milella, A., 2023. Automated detection and counting of grape bunches using a farmer robot.

Eli-Chukwu, N.C., 2019. Applications of artificial intelligence in agriculture: A review. Eng. Technol. Appl. Sci. Res. 9 (4).

Gan, Z., Li, L., Li, C., Wang, L., Liu, Z., Gao, J., et al., 2022. Vision-language pre-training: Basics, recent advances, and future trends. Found. Trends® Comput. Graph. Vis. 14 (3–4), 163–352.

Ghoury, S., Sungur, C., Durdu, A., 2019. Real-time diseases detection of grape and grape leaves using faster r-cnn and ssd mobilenet architectures. In: International Conference on Advanced Technologies, Computer Engineering and Science. ICATCES 2019, pp. 39–44.

Grazioso, A., Ugenti, A., Galati, R., Mantriota, G., Reina, G., 2023. Modeling and validation of a novel tracked robot via multibody dynamics. Robotica 41 (10), 3211–3232.

Kalyan, K.S., Rajasekharan, A., Sangeetha, S., 2021. Ammus: A survey of transformer-based pretrained models in natural language processing. ArXiv preprint arXiv:2108.05542.

Kamilaris, A., Prenafeta-Boldú, F.X., 2018. A review of the use of convolutional neural networks in agriculture. J. Agric. Sci. 156 (3), 312–322.

Khan, S., Naseer, M., Hayat, M., Zamir, S.W., Khan, F.S., Shah, M., 2022. Transformers in vision: A survey. ACM Comput. Surv. (CSUR) 54 (10s), 1–41.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.-Y., et al., 2023. Segment anything. ArXiv preprint arXiv:2304.02643.

Lin, G., Tang, Y., Zou, X., Xiong, J., Li, J., 2019. Guava detection and pose estimation using a low-cost RGB-D sensor in the field. Sensors 19 (2).

Lin, T., Wang, Y., Liu, X., Qiu, X., 2022. A survey of transformers. AI Open.

Liu, X., 2023. A SAM-based method for large-scale crop field boundary delineation. In: 2023 20th Annual IEEE International Conference on Sensing, Communication, and Networking. SECON, pp. 1–6. http://dx.doi.org/10.1109/SECON58729.2023.10287502.

Liu, B., Luo, L., Wang, J., Lu, Q., Wei, H., Zhang, Y., Zhu, W., 2023. An improved lightweight network based on deep learning for grape recognition in unstructured environments. Inf. Process. Agric..

Luo, L., Tang, Y., Zou, X., Ye, M., Feng, W., Li, G., 2016. Vision-based extraction of spatial information in grape clusters for harvesting robots. Biosyst. Eng. 151, 90–104.

Milella, A., Marani, R., Petitti, A., Reina, G., 2019. In-field high throughput grapevine phenotyping with a consumer-grade depth camera. Comput. Electron. Agric. 156, 293–306.

Najafabadi, M.M., Villanustre, F., Khoshgoftaar, T.M., Seliya, N., Wald, R., Muharemagic, E., 2015. Deep learning applications and challenges in big data analytics. J. Big Data 2 (1), 1–21.

Osco, L.P., Wu, Q., de Lemos, E.L., Gonçalves, W.N., Ramos, A.P.M., Li, J., Junior, J.M., 2023. The segment anything model (sam) for remote sensing applications: From zero to one shot. Int. J. Appl. Earth Obs. Geoinf. 124, 103540.

Padilla, R., Netto, S.L., Da Silva, E.A., 2020. A survey on performance metrics for object-detection algorithms. In: 2020 International Conference on Systems, Signals and Image Processing. IWSSIP, IEEE, pp. 237–242.

Saiz-Rubio, V., Rovira-Más, F., 2020. From smart farming towards agriculture 5.0: A review on crop data management. Agronomy 10 (2), 207.

Sharma, A., Jain, A., Gupta, P., Chowdary, V., 2020. Machine learning applications for precision agriculture: A comprehensive review. IEEE Access 9, 4843–4873.

Shen, L., Su, J., He, R., Song, L., Huang, R., Fang, Y., Song, Y., Su, B., 2023. Real-time tracking and counting of grape clusters in the field based on channel pruning with YOLOv5s. Comput. Electron. Agric. 206, 107662.

Shen, L., Su, J., Huang, R., Quan, W., Song, Y., Fang, Y., Su, B., 2022. Fusing attention mechanism with mask R-CNN for instance segmentation of grape cluster in the field. Front. Plant Sci. 13, 934450.

Singh, R.S.R., Sanodiya, R.K., 2023. Zero-shot transfer learning framework for plant leaf disease classification. IEEE Access.

Tan, X., Xi, B., Li, J., Zheng, T., Li, Y., Xue, C., Chanussot, J., 2024. Review of zero-shot remote sensing image scene classification. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens..

Van Klompenburg, T., Kassahun, A., Catal, C., 2020. Crop yield prediction using machine learning: A systematic literature review. Comput. Electron. Agric. 177, 105709.

Wang, D., Zhang, J., Du, B., Tao, D., Zhang, L., 2023. Scaling-up remote sensing segmentation dataset with segment anything model. ArXiv preprint arXiv:2305.02034.

Wang, W., Zheng, V.W., Yu, H., Miao, C., 2019. A survey of zero-shot learning: Settings, methods, and applications. ACM Trans. Intell. Syst. Technol. 10 (2), 1–37.

Williams, D., Macfarlane, F., Britten, A., 2024. Leaf only SAM: A segment anything pipeline for zero-shot automated leaf segmentation. Smart Agric. Technol. 8, 100515.

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.-Y., 2022. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. ArXiv preprint arXiv:2203.03605.

Zhang, C., Marfatia, P., Farhan, H., Di, L., Lin, L., Zhao, H., Li, H., Islam, M.D., Yang, Z., 2023. Enhancing USDA nass cropland data layer with segment anything model. In: 2023 11th International Conference on Agro-Geoinformatics. Agro-Geoinformatics, pp. 1–5. http://dx.doi.org/10.1109/Agro-Geoinformatics59224.2023.10233404.

Zhong, F., Chen, Z., Zhang, Y., Xia, F., 2020. Zero-and few-shot learning for diseases recognition of citrus aurantium L. using conditional adversarial autoencoders. Comput. Electron. Agric. 179, 105828.