

AgriSORT: A Simple Online Real-time Tracking-by-Detection framework for robotics in precision agriculture

Leonardo Saraceni, Ionut M. Motoi, Daniele Nardi, Thomas A. Ciarfuglia

Abstract—The problem of multi-object tracking (MOT) consists in detecting and tracking all the objects in a video sequence while keeping a unique identifier for each object. It is a challenging and fundamental problem for robotics. In precision agriculture the challenge of achieving a satisfactory solution is amplified by extreme camera motion, sudden illumination changes, and strong occlusions. Most modern trackers rely on the appearance of objects rather than motion for association, which can be ineffective when most targets are static objects with the same appearance, as in the agricultural case. To this end, on the trail of SORT [5], we propose AgriSORT, a simple, online, real-time tracking-by-detection pipeline for precision agriculture based only on motion information that allows for accurate and fast propagation of tracks between frames. The main focuses of AgriSORT are efficiency, flexibility, minimal dependencies, and ease of deployment on robotic platforms. We test the proposed pipeline on a novel MOT benchmark specifically tailored for the agricultural context, based on video sequences taken in a table grape vineyard, particularly challenging due to strong self-similarity and density of the instances. Both the code and the dataset are available for future comparisons at: <https://github.com/Lio320/AgriSORT>

I. INTRODUCTION

The deployment of robots in the precision agriculture context (Fig. 1) has seen rapid growth because of their potential to reduce the high cost of repetitive labor and wasted resources. Some possible tasks are weeding, fruit detection and harvesting, yield estimation, and fertilizer or pesticide application. Many works [20, 37] in literature have addressed the problem of detecting crops in single frames using various sensors. Although analyzing individual frames can offer insight into a crop, this is not enough for many tasks. In fact, since changing point of view may lead to an entirely different measurement, algorithms need to be able to integrate information from multiple frames, updating temporal information in real-time. In the literature, many works provide use cases in which it is crucial to have an efficient and reliable MOT algorithm. These include:

- **Efficient use of resources:** By tracking individual plants, farmers can optimize resources such as water,

■ This work has been partially supported by the European Commission under the grant agreement number 101016906 – Project CANOPIES

■ This work has been partially supported by project AGRITECH Spoke 9 - Codice progetto MUR: AGRITECH "National Research Centre for Agricultural Technologies" - CUP CN00000022, of the National Recovery and Resilience Plan (PNRR) financed by the European Union "Next Generation EU".

This work has been partially supported by Sapienza University of Rome as part of the work for project *H&M: Hyperspectral and Multispectral Fruit Sugar Content Estimation for Robot Harvesting Operations in Difficult Environments*, Del. SA n.36/2022.



Fig. 1: Agricultural robotic platform used in the EU Project CANOPIES for operations in table grape vineyards, equipped with an Intel RealSense d435i camera on the wrist. AgriSORT is implemented on this robot, and is used for tracking grapes.

fertilizer, and pesticides, saving costs and reducing the environmental impact of agriculture [36].

- **Crop management:** MOT can be combined with other computer vision techniques in the field to make real-time decisions. Some of those include the identification of weeds to apply pesticides and herbicides [6], precision spraying [18], and estimation of the quality of crops.
- **Yield estimation:** For yield estimation, tracking approaches are necessary to ensure objects are only counted once [16][10]
- **Improved accuracy:** Multiple object tracking technology can help agricultural robots to accurately track each plant, even in challenging conditions, such as when plants are out of the camera's field of view [19]. In particular, tracking can provide more consistent results than detection-only applications for problems like quality assessment.

Multi-object tracking (MOT) is a computer vision problem that aims to identify and locate objects of interest in a video sequence, to associate them across frames to keep track of their movements over time. MOT is a fundamental issue for various applications, one of which is precision agriculture. Most SOTA tracking methods, such as SORT [5], DeepSORT [42], or JDE [41], use approaches based on appearance models since they are designed to track easily distinguishable moving objects from a still or slow-moving camera. While this approach is very effective in the context of classical MOT targets such as cars or people [15, 23, 12], for which

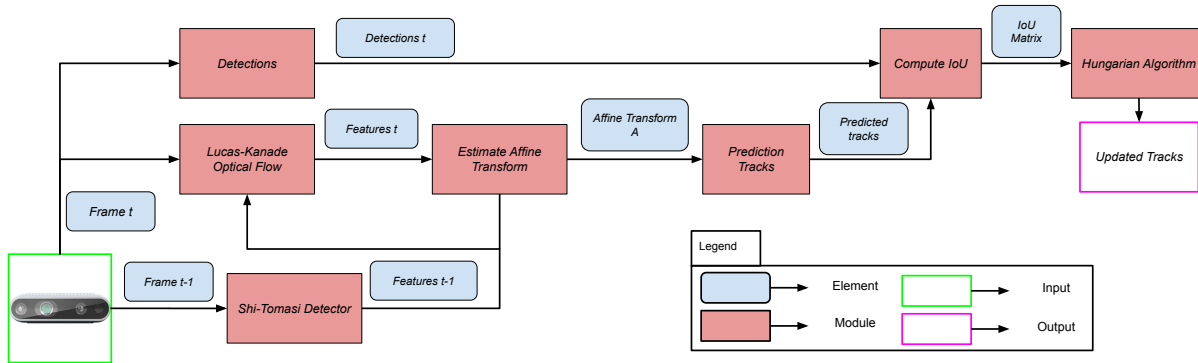


Fig. 2: Overview of our AgriSORT tracker pipeline. The process starts with the estimation of the relative camera motion. At first we extract features using the Shi-Tomasi method in the previous and current frame to compute the Optical Flow via the Lucas-Kanade algorithm. Using the generated matches we estimate an affine transform that expresses the motion of the camera. The estimated matrix is then used to propagate the state of the previous tracks in the current frame. At the end we compute the IoU with the current detection to perform the association and update the state of the tracklets.

it is possible to extract reliable appearance features, we found that they tend to be less effective when applied to the agricultural context. The main reason behind this is that in a farm environment, all crops are visually similar and static. In addition, appearance models are trained on additional data, besides those used to train the detection model, that is often unavailable in agricultural contexts, where data is scarce and often requires experts for labelling. To address these issues, we present AgriSORT, a perception pipeline incorporating the detection and tracking of crops, designed to be used on a robotic platform. Since our method does not use appearance features to identify instances in consecutive frames, we solve the association problem by using a Kalman Filter formulation different from the usual MOT algorithms, designed to follow the rapid camera movements with higher performance. In addition, the proposed approach is general and easily extendable to any crop type for which a working detector is available. The contribution of this work can be summarized as follows:

- A MOT tracking solution especially suited for agricultural robotics scenarios is proposed.
- A novel agricultural MOT benchmark, based on sequences collected on a table grape vineyard, is presented.
- The effectiveness of the proposed solution is experimentally evaluated and compared with the recent state-of-the-art MOT solutions.

II. RELATED WORKS

A. Vision-based application for precision agriculture

Recently, robotic systems with different degrees of autonomy are spreading in agriculture to support farmers in various operations. LIDARs and GPS [26, 40] are among the most used sensors for these applications. However, these systems are costly and impose significant constraints from the point of view of the overall weight, the power used, and the availability of a fixed reference system on the ground. Therefore, the interest in vision strategies has grown over the years as they require only a camera of low weight and low cost. Many works in literature [17, 27, 34] propose methods to solve the problem of detecting crops using hand-crafted

features. However, they require careful tuning to achieve good domain generalization, so more recent approaches use the power of Deep Neural Network (DNN) detectors to detect crops. To be more specific, FasterRCNN has been deployed for the detection of apples [31], mangoes [2], tomatoes [35], and oranges [31]. Other works employed the YOLOv3 [25] architecture or the YOLOv5 for detecting tomatoes [45].

B. Multiple Object Tracking

In the MOT literature, methods are classified as separate and joint trackers. Separate trackers [5, 42, 14, 43, 13, 7, 44, 18] follow the tracking-by-detection paradigm, which localizes and classifies all targets first and then associates detections belonging to the same object using information on appearance, motion, or both. On the other hand, joint trackers [3, 41, 46, 39, 38, 32, 24, 28] unify the detection and appearance model into a single framework and train them together. Compared to separate trackers, the main advantage of joint methods is their high inference speed and comparable performance in simple scenes. However, they tend to fail in more sophisticated scenarios. The closest work in motivation to ours is LettuceTrack [18], which proposes a tracking pipeline for the detection of lettuce using a wheeled robotic platform with an RGB camera pointed downwards, while the robot is moving in a straight line. The association problem is solved with geometrical considerations by exploiting the regular pattern of the planted crops. For these reasons, their approach is not effective for unconstrained motion and variable illumination conditions, typical of robotic monitoring applications, while our approach has a much wider applicability. We demonstrate it on the case of table grape cultivation, which has specific challenges of instance similarity, instances distance and separability, and rapid illumination changes.

III. MATERIALS AND METHODS

In this section we describe the data collection and give a detailed description of the tracking pipeline. In Section III-A the experimental field and the data collection and labelling are described. In Sections III-B and III-C we discuss the Kalman Filter formulation and the association process. The algorithm pipeline is presented in Fig 2.



Fig. 3: Experimental setup for data acquisition, a Intel RealSense d435i camera mounted on a tripod for stabilization during movements.

A. Data acquisition

The dataset used in this paper has been collected in a vineyard of table grapes located in southern Lazio (Italy). The vineyards are structured as a traditional trellis system called Tendone, with a vast distance between each plant of 3 meters. Plantations are all in full production and health, thus representing a typical working condition for validating agromonic activities. This work is part of an extensive agricultural robotics project called Canopies (Fig. 1). For this reason, we used the project’s hardware to collect data, consisting of a depth camera, Intel RealSense D435i (Fig. 3). The sequences were acquired while moving along the vineyard rows and pointing toward the grape bunches. Due to the specifications of the projects, even though the camera’s full resolution is FullHD (1920x1080), we recorded the data with HD resolution (1280x720) at 30 FPS to allow for alignment between depth and RGB images. This work doesn’t use the acquired depth to keep the hardware requirements more straightforward, but we will use it in the future for other works. We downsampled some sequences to 10 FPS to see how the detector works at a lower framerate. During the data acquisition campaign, the images are collected with the handheld camera, simulating the motion of the robot during operations like harvesting or yield estimation. We manually annotated the collected data to provide a test set to validate the performance of the proposed method using Computer Vision Annotation Tool [11] with the MOT format.

TABLE I: List of sequences acquired and labelled for MOT with their characteristics

Dataset	CloseUp1	CloseUp2	Overview1	Overview2
Resolution	1280x720	1280x720	1280x720	1280x720
Length (frames)	300	300	100	100
FPS	30	30	10	10
Tracks	23	22	20	31
Boxes	2583	3581	1040	721

The sequences involve various movements within the vineyard rows, such as sudden changes in direction, U-shaped patterns, and close-up shots of table grape bunches. Details of the dataset are summarized in Table I. The analysis of table grapes for the purposes of robotic agriculture presents intriguing but formidable challenges due to the relatively unstructured nature of the environment. Unlike crops such as lettuce or cabbage that are planted in fixed patterns, table

grapes grow along rows without any specific arrangement. This poses a challenge for tracking, as occlusions occur frequently due to overlapping crops, dense foliage, and poles. The dataset for this tracking task is unique, as the objects being tracked remain static while the camera moves relatively to the crops. This task is made difficult by irregular and unpredictable (human or robotic) motion, which can be very fast. Additionally, the agricultural environment is highly unstructured, and the terrain is steep, resulting in noise from camera vibrations and minor oscillations. Extreme movements due to potholes or other obstructions also cause motion blur. Illumination changes rapidly and can sometimes be directed toward the camera, causing unusable frames. Some examples of difficulties can be found in Fig. 4.

B. Kalman Filter formulation

Each of the detected objects is instantiated with a Kalman Filter (KF) that tracks its geometrical properties through time. There are a number of facts to consider when designing the KF for bounding box tracking in the context of agricultural scenarios. Since robot motion along the field happens on a rough terrain, the camera is subject to huge vibrations that translate to unpredictable motion in the image coordinates. It is common practice in MOT to include in the KF state vector the derivatives of the bounding box size and motion, and this works well when the hypothesis of linear motion holds. Since this is not the case, the derivative of motion and size become detrimental to the KF estimation. A second consideration is that, given camera high frame rates compared to its velocity, it is easy to detect rapid direction changes and to approximate the camera motion using simple affine transformations. Therefore, relying on the motion vector for correction of the state vector, instead of using a transition matrix of any sort, is a simple and effective solution to the tracking problem in this context.

The state and measurement vectors are defined as follows:

$$x_k = [x_c(k), y_c(k), w(k), h(k)]^T \quad (1)$$

$$z_k = [z_{x_c}(k), z_{y_c}(k), z_w(k), z_h(k)]^T \quad (2)$$

where k is the time step, (w, h) are the width and height of the bounding box, \mathbf{x} is the state vector, and \mathbf{z} is the measurement vector.

To estimate the camera motion we use Lucas-Kanade sparse optical flow correspondences [29] and solve for an affine transformation matrix, defined as follows:

$$A_{k-1}^k = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \quad (3)$$

The affine transform parameters represent a linear mapping between points in the reference and current images, describing how to transform the reference image to align it with the current image. In particular the translation components (a_{13}, a_{23}) provide an estimate of the camera’s displacement in the x and y directions, while the scale and rotation components $(a_{11}, a_{12}, a_{21}, a_{22})$ can indicate changes in scale and rotation.



Fig. 4: Example of difficult cases present in the experimental field. Strong frontal illumination and motion blur due to fast motion (a), and occlusions due to leaves and branches (b).

Affine motion can capture translation, rotation, scaling, and shearing effects but not perspective distortions. Since most of the robot motion is parallel to the fruit orchard rows, the perspective effects are limited. In addition, the high frame rate further reduces the error due to this approximation. The performance advantages of using simple affine transformation will be discussed in Section IV. Given the affine transform, the custom prediction step updates only the center of the bounding box, projecting it between frames and keeping width and height fixed. Therefore the propagation step follows Eq.(4).

$$\begin{bmatrix} x_c(k) \\ y_c(k) \\ w(k) \\ h(k) \end{bmatrix} = \begin{bmatrix} a_{11} \cdot x_c(k-1) + a_{12} \cdot y_c(k-1) + a_{13} \\ a_{21} \cdot x_c(k-1) + a_{22} \cdot y_c(k-1) + a_{23} \\ w(k-1) \\ h(k-1) \end{bmatrix} \quad (4)$$

C. Association

For what concerns the association step, given a set of observations at frame t and a set of predicted observations according to the estimation model, we aim to find the optimal assignment of observations to predicted states.

We formalize the problem as an optimization problem over an assignment matrix A , where each element (A_{ij}) indicates the intersection over union (IOU) between the i_{th} observation and the j_{th} predicted state. Specifically, $A_{ij} = 1$ if the i_{th} observation perfectly overlaps with the j_{th} prediction, while $A_{ij} = 0$ means the two objects do not overlap. To find the optimal assignment between observations and predicted states that maximizes the overall IOU, we use the Hungarian algorithm [22], a well-known solution for these problems.

IV. EXPERIMENTS

In the following section, we explain the implementation details of the proposed method as well as the evaluation metrics to test the performance and the comparison with other state-of-the-art MOT methods.

A. Implementation details

The detector employed for this application is the YOLO-V5s model of the YOLO-V5 family because it provides the best trade-off between inference speed and accuracy, which is vital for real-time tracking applications. We trained the

model on the table-grapes dataset presented in [9, 8], which consists of 242 annotated images and 1469 images automatically annotated using a pseudo-label generation strategy. The training details can be found in [9, 8]. The NVIDIA GeForce RTX 3070 Ti Laptop GPU is the inference and training hardware. The choice for the system and measurement noise matrices Q_k and R_k , respectively, consist of 4x4 diagonal matrices with fixed noise factors, which are $\sigma_q = 0.05$ and $\sigma_r = 0.00625$, also multiplied by a factor (δt) that depends on the framerate of the camera sequence (0.033 for the 30 FPS sequences, and 0.1 for the 10 FPS sequences)

$$\begin{aligned} Q_k &= \text{diag}_{4 \times 4}((\sigma_q)^2) \cdot \delta t \\ R_k &= \text{diag}_{4 \times 4}((\sigma_r)^2) \cdot \delta t \end{aligned} \quad (5)$$

The confidence threshold used for the detector is 0.4, with a NMS value of 0.5. We consider a track "lost" when it is not associated with any detection with more than ten frames. At the same time, we "insert" a new track when detection is associated with more than three consecutive times to filter out fake detections.

B. Baselines

To compare with other methods, we choose some baseline SOTA MOT approaches. In particular, we choose SORT because it is the first and most straightforward tracking-by-detection approach. Additionally, we selected ByteTrack [43] and OC-SORT [7] because of their methodological solid innovations and the integration of a camera motion compensation module provided by OpenCV into the pipeline. We also compare with StrongSORT [14] and BoTSORT [1] since both achieved excellent results on pedestrian datasets MOT17 and MOT20 [12]. We shared the same detector model for each tracker to provide a fair comparison with the other state-of-the-art trackers. However, some methods (StrongSORT and BoTSORT) require an additional model for appearance feature extraction besides the detector, which we cannot finetune due to the absence of data. Therefore the pre-trained default models provided by the authors are used. The benchmark parameters of the baselines are the same as those in AgriSORT where possible to provide a fair comparison. For all the other parameters we refer to the original papers for more details.

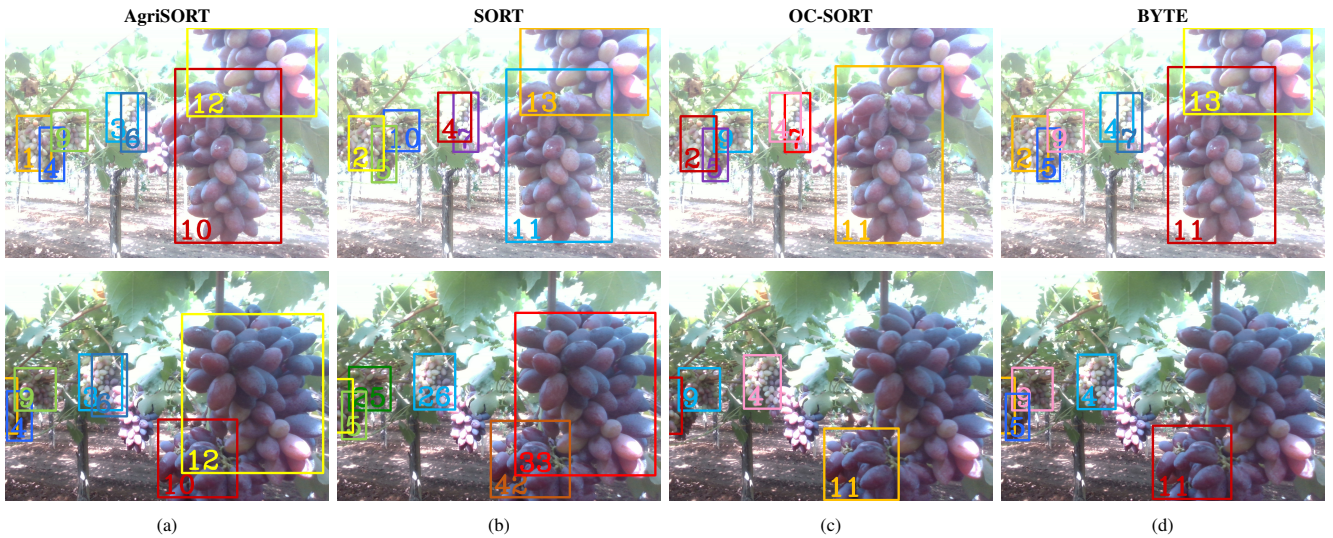


Fig. 5: Qualitative evaluation of the performance of Agrisort and other SOTA trackers in the “CloseUp1” sequence at frame 20 (top row) and 80 (bottom row). Agrisort (a) is able to keep consistency on most tracks, without switching IDs and providing the least number of false negatives. SORT (b) suffers in this sequence, it loses almost all the IDs, due to the presence of fast non-linear motion and illumination changes. OC-SORT (c) and BYTE (d) perform better than SORT, in particular without switching IDs, however their performance is far from Agrisort because they lose some of the tracks both in foreground and background.

TABLE II: Comparison of Agrisort performance with other tracking-by-detection trackers on agricultural sequences

Sequence	Method	MOTA \uparrow	IDF1 \uparrow	HOTA \uparrow	FP \downarrow	FN \downarrow	IDs \downarrow	MT \uparrow	ML \downarrow
CloseUp1	BOTSORT [1]	48.66	72.72	40.51	197	1381	56	4	7
	BYTETRACK [43]	52.19	56.01	37.153	51	1157	27	4	7
	OC-SORT [7]	48.70	62.27	40.52	37	1272	16	2	7
	SORT [5]	48.238	44.27	31.43	31	1271	35	2	7
	StrongSORT [14]	57.37	60.25	41.20	21	1054	26	6	7
	Agrisort (ours)	65.93	72.00	48.71	459	872	11	11	4
CloseUp2	BOTSORT [1]	51.58	61.645	47.02	405	1805	30	8	4
	BYTETRACK [43]	53.81	64.42	46.16	371	1703	18	8	3
	OC-SORT [7]	48.42	62.95	45.99	233	1829	9	6	5
	SORT [5]	47.11	55.41	43.48	424	2046	38	6	6
	StrongSORT [14]	56.46	55.19	43.16	371	1703	18	8	3
	Agrisort (ours)	66.13	73.00	56.08	655	1146	10	12	1
Overview1	BOTSORT [1]	48.46	64.77	40.32	244	425	53	5	1
	BYTETRACK [43]	55.58	62.94	41.48	248	448	29	7	2
	OC-SORT [7]	53.94	69.43	42.24	143	411	15	5	2
	SORT [5]	62.69	73.91	51.20	108	367	9	8	2
	StrongSORT [14]	48.94	62.29	38.81	346	413	22	8	0
	Agrisort (ours)	62.21	73.72	52.74	255	284	12	13	0
Overview2	BOTSORT [1]	30.652	50.193	32.434	249	396	55	8	6
	BYTETRACK [43]	42.302	54.784	35.423	220	366	20	10	5
	OC-SORT [7]	41.748	55.783	34.578	158	381	11	6	5
	SORT [5]	49.515	61.96	44.431	110	348	12	10	8
	StrongSORT [14]	34.258	54.093	34.28	304	341	17	12	6
	Agrisort (ours)	45.08	56.88	41.33	298	316	14	16	4

C. Evaluation metrics

Evaluations were performed according to a combination of widely accepted metrics defined by [33, 30, 4]. Those include Multiple Object Tracking Accuracy (MOTA), False Positive (FP), False Negative (FN), ID Switch (IDs), IDF1, Higher-Order Tracking Accuracy (HOTA), Mostly Tracked (MT), and Mostly Loss (ML). MOTA is a metric commonly used to assess the performance of MOT algorithms. It measures the overall tracking accuracy by considering false positives (FP) and false negatives (FN). MOTA focuses more on detection performance because the amount of FP and FN are more

significant than the IDs. On the other hand, IDF1 is a metric that evaluates the ability of the tracker to correctly identify individual objects across frames, combining precision and recall of identity assignments. HOTA is an extension of MOTA that incorporates additional metrics to evaluate the tracking performance at different levels, including localization, classification, and association. FP refers to the number of objects incorrectly detected as present when they are not, while FN refers to the number of targets missed or not detected by the tracking algorithm when they are present in the scene. MT and ML are binary metrics indicating whether an object is mostly tracked or lost throughout the

sequence. An object is considered mostly tracked (MT) if its overlap with the ground truth is above a certain threshold for a significant portion of its lifespan. To compute the above metrics, we use TrackEval [21], the official reference implementation for the HOTA metrics.

D. Results and discussions

The performance of the proposed method and compared to the baselines are summarized in Table II. It can be seen that on the *CloseUp* sequences our method always outperforms all the others. The fact that those sequences are mostly characterized by occlusions, irregular motion, and illumination changes, proves that our method is robust to those criticalities. OC-SORT provides the best results in terms of "IDswitch," but it also provides the highest number of false negatives, meaning that it ignores a large number of instances and so is not suitable for agronomic operations. In the *Overview* sequences our approach consistently exhibits stronger capabilities compared to the other trackers, with a few exceptions where SORT remains competitive in some metrics by a small margin. This happens because SORT excels particularly when the assumption of linear motion holds, as in the *Overview2* sequence, because it consists of a slow and regular walk parallel to the vineyard. However, this assumption is overly restrictive and does not accurately represent real-world robotics scenarios, where robots must move and approach crops closely to perform various operations. The results also show that AgriSORT consistently outperforms all its competitors in terms of MT and ML tracks, underscoring its robust association capabilities. These metrics are important for agronomic tasks like yield estimation, where it is important to not lose tracks, otherwise the same grape bunch gets counted more than once, leading to a wrong estimate of the workload.

Regarding overall results, we expected BoT-SORT and StrongSORT to perform poorly because they use deep models to extract visual information from objects that have not been finetuned on our data. Moreover, as stated in [18], tracking plants presents distinct challenges compared to tracking humans due to the inherent similarities between different instances and the absence of distinct characteristics for disambiguation. In contrast, our method exhibits greater resilience by relying solely on easily extractable motion information within a static environment. In the comparison with OC-SORT, SORT, and ByteTrack, all of which rely solely on the detector, our association strategy excels within our specific context, primarily due to its robustness in handling abrupt motion changes. Visual representations of the performance disparities among these various trackers can be found in Fig. 5.

We also conducted a study to compare different strategies for estimating the camera motion. In particular, we explored the possibility of substituting the affine transform with a homography matrix and the Lucas-Kanade Optical Flow method with the ORB feature matching. The additional investigation results are displayed in Table III and show that there is no clear trend demonstrating the dominance of a

TABLE III: Comparison of the performance of different techniques to estimate the camera motion.

Sequence	Technique	MOTA \uparrow	IDF1 \uparrow	HOTA \uparrow	FPS \uparrow
CloseUp1	LK + Aff	65.93	72.00	48.71	25.93
	ORB + Aff	66.59	72.83	52.55	20.92
	LK + hom	64.85	71.53	51.90	25.80
	ORB + hom	67.28	76.32	53.80	19.71
CloseUp2	LK + Aff	66.13	73.00	56.08	66.09
	ORB + Aff	66.32	73.10	56.12	25.19
	LK + hom	64.59	74.99	57.24	63.01
	ORB + hom	65.48	74.64	56.64	25.01
Overview1	LK + Aff	62.21	73.72	52.74	55.99
	ORB + Aff	62.20	74.05	53.14	23.93
	LK + hom	60.19	70.55	49.85	53.36
	ORB + hom	60.48	71.03	51.54	22.79
Overview2	LK + Aff	45.08	56.88	41.33	55.43
	ORB + Aff	41.75	54.53	38.75	25.42
	LK + hom	45.08	54.66	40.27	50.35
	ORB + hom	39.39	57.12	40.39	24.75

single technique over the others in terms of metrics, and that the results depend on the characteristics of the single sequence. Despite the minimal differences, the affine transform for motion estimation is superior in the "Overview" sequences compared to homography because the motion is mainly linear and parallel to the crops, which corresponds to small scale changes. On the other hand, homography works better in the "CloseUp" sequences, because they are characterized by the toughest motion, including fast lateral motion and scale changes. Since the metrics are similar, the most important parameter to consider is the computational speed in Frames per Second (FPS) indicating the speed the algorithm takes to process the input, which is vital in robotics applications. The time indicated in the table does not include the detection processing time since it is the same for all techniques and is about 10 ms. The results show that the ORB feature extraction method is much slower than the LK method by an average of 34 FPS in almost all sequences except for the *CloseUp1* sequence, where the speed difference shrinks to only 5 FPS due to some singularities. In our implementation we provide the possibility to change the techniques as parameters, but we propose the combination of affine transform and Lucas-Kanade method for feature extraction as our main pipeline since it is the faster method.

V. CONCLUSION

In this paper, we present AgriSORT, a MOT approach for robotics in precision agriculture, light, fast, and flexible enough that it can be applied to different types of crops given a working detection algorithm. We propose a different formulation of the Kalman Filter specific for the agricultural case, where objects of interest are static and the only source of motion is the dynamic camera. We conducted experiments on real data collected and annotated to validate our method. We compared AgriSORT to other SOTA approaches for MOT to demonstrate that, in this context, it performs better while running at high-speed, confirming its real-time deployment. Future directions include improving the method for estimating the camera motion using learning-based techniques and extending the tracker to mixed classes of objects, including moving ones like people or tractors, which are helpful from an operational perspective.

REFERENCES

- [1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. “BoT-SORT: Robust associations multi-pedestrian tracking”. In: *arXiv preprint arXiv:2206.14651* (2022).
- [2] Suchet Bargoti and James Underwood. “Deep fruit detection in orchards”. In: *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 3626–3633.
- [3] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. “Tracking without bells and whistles”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 941–951.
- [4] Keni Bernardin and Rainer Stiefelhagen. “Evaluating multiple object tracking performance: the clear mot metrics”. In: *EURASIP Journal on Image and Video Processing* 2008 (2008), pp. 1–10.
- [5] Alex Bewley et al. “Simple online and realtime tracking”. In: *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, Sept. 2016. DOI: 10.1109/icip.2016.7533003. URL: <https://doi.org/10.11092Ficip.2016.7533003>.
- [6] Andrea Botta et al. “A Review of Robots, Perception, and Tasks in Precision Agriculture”. In: *Applied Mechanics* 3.3 (2022), pp. 830–854. ISSN: 2673-3161. DOI: 10.3390/applmech3030049. URL: <https://www.mdpi.com/2673-3161/3/3/49>.
- [7] Jinkun Cao et al. *Observation-Centric SORT: Rethinking SORT for Robust Multi-Object Tracking*. 2023. arXiv: 2203.14360 [cs.CV].
- [8] Thomas A Ciarfuglia et al. “Pseudo-label generation for agricultural robotics applications”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 1686–1694.
- [9] Thomas A Ciarfuglia et al. “Weakly and semi-supervised detection, segmentation and tracking of table grapes with limited and noisy data”. In: *Computers and Electronics in Agriculture* 205 (2023), p. 107624.
- [10] Thomas A. Ciarfuglia et al. “Weakly and semi-supervised detection, segmentation and tracking of table grapes with limited and noisy data”. In: *Computers and Electronics in Agriculture* 205 (2023), p. 107624. ISSN: 0168-1699. DOI: <https://doi.org/10.1016/j.compag.2023.107624>. URL: <https://www.sciencedirect.com/science/article/pii/S0168169923000121>.
- [11] CVAT.ai Corporation. *Computer Vision Annotation Tool (CVAT)*. Version 2.2.0. Sept. 2022. URL: <https://github.com/opencv/cvat>.
- [12] P. Dendorfer et al. “MOT20: A benchmark for multi object tracking in crowded scenes”. In: *arXiv:2003.09003[cs]* (Mar. 2020). arXiv: 2003.09003. URL: <http://arxiv.org/abs/1906.04567>.
- [13] Yunhao Du et al. “GiaoTracker: A comprehensive framework for mcmot with global information and optimizing strategies in visdrone 2021”. In: *Proceedings of the IEEE/CVF International conference on computer vision*. 2021, pp. 2809–2819.
- [14] Yunhao Du et al. “Strongsort: Make deepsort great again”. In: *IEEE Transactions on Multimedia* (2023).
- [15] Andreas Geiger, Philip Lenz, and Raquel Urtasun. “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2012.
- [16] Michael Halstead et al. “Crop Agnostic Monitoring Driven by Deep Learning”. In: *Frontiers in Plant Science* 12 (Dec. 2021). DOI: 10.3389/fpls.2021.786702.
- [17] Sebastian Haug et al. “Plant classification system for crop/weed discrimination without segmentation”. In: *IEEE winter conference on applications of computer vision*. IEEE, 2014, pp. 1142–1149.
- [18] Nan Hu et al. “LettuceTrack: Detection and tracking of lettuce for robotic precision spray in agriculture”. In: *Frontiers in Plant Science* 13 (2022). ISSN: 1664-462X. DOI: 10.3389/fpls.2022.1003243. URL: <https://www.frontiersin.org/articles/10.3389/fpls.2022.1003243>.
- [19] Yuhua Ji et al. “Multiple object tracking in farmland based on fusion point cloud data”. In: *Computers and Electronics in Agriculture* 200 (2022), p. 107259. ISSN: 0168-1699. DOI: <https://doi.org/10.1016/j.compag.2022.107259>. URL: <https://www.sciencedirect.com/science/article/pii/S0168169922005725>.
- [20] Xiaojun Jin et al. “A novel deep learning-based method for detection of weeds in vegetables”. In: *Pest Management Science* 78.5 (2022), pp. 1861–1869.
- [21] Arne Hoffhues Jonathon Luiten. *TrackEval*. <https://github.com/JonathonLuiten/TrackEval>. 2020.
- [22] Harold W Kuhn. “The Hungarian method for the assignment problem”. In: *Naval research logistics quarterly* 2.1-2 (1955), pp. 83–97.
- [23] L. Leal-Taixé et al. “MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking”. In: *arXiv:1504.01942 [cs]* (Apr. 2015). arXiv: 1504.01942. URL: <http://arxiv.org/abs/1504.01942>.
- [24] Jiaxin Li et al. “Simpletrack: Rethinking and improving the jde approach for multi-object tracking”. In: *Sensors* 22.15 (2022), p. 5863.
- [25] Guoxu Liu et al. “YOLO-tomato: A robust algorithm for tomato detection based on YOLOv3”. In: *Sensors* 20.7 (2020), p. 2145.
- [26] Tianhao Liu, Hanwen Kang, and Chao Chen. “ORB-Livox: A real-time dynamic system for fruit detection and localization”. In: *Computers and Electronics in Agriculture* 209 (2023), p. 107834.
- [27] Philipp Lottes et al. “UAV-based crop and weed classification for smart farming”. In: *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 3024–3031.
- [28] Zhichao Lu et al. “Retinatrack: Online single stage joint detection and tracking”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 14668–14678.
- [29] Bruce D Lucas and Takeo Kanade. “An iterative image registration technique with an application to stereo vision”. In: *IJCAI’81: 7th international joint conference on Artificial intelligence*. Vol. 2. 1981, pp. 674–679.
- [30] Jonathon Luiten et al. “Hota: A higher order metric for evaluating multi-object tracking”. In: *International journal of computer vision* 129 (2021), pp. 548–578.
- [31] Xiaochun Mai et al. “Faster R-CNN with classifier fusion for automatic detection of small fruits”. In: *IEEE Transactions on Automation Science and Engineering* 17.3 (2020), pp. 1555–1569.
- [32] Tim Meinhardt et al. “Trackformer: Multi-object tracking with transformers”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 8844–8854.
- [33] Anton Milan et al. “MOT16: A benchmark for multi-object tracking”. In: *arXiv preprint arXiv:1603.00831* (2016).
- [34] Andres Milioto, Philipp Lottes, and Cyrill Stachniss. “Real-time blob-wise sugar beets vs weeds classification for monitoring fields using convolutional neural networks”. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 4 (2017), pp. 41–48.
- [35] Yue Mu et al. “Intact detection of highly occluded immature tomatoes on plants using deep learning techniques”. In: *Sensors* 20.10 (2020), p. 2984.
- [36] Youssef Osman, Reed Dennis, and Khalid Elgazzar. “Yield Estimation and Visualization Solution for Precision Agriculture”. In: *Sensors* 21.19 (2021). ISSN: 1424-8220. DOI: 10.3390/s21196657. URL: <https://www.mdpi.com/1424-8220/21/19/6657>.
- [37] Muhammad Hammad Saleem, Johan Potgieter, and Khalid Mahmood Arif. “Automation in agriculture by machine and deep learning techniques: A review of recent developments”. In: *Precision Agriculture* 22 (2021), pp. 2053–2091.
- [38] Peize Sun et al. “Transtrack: Multiple object tracking with transformer”. In: *arXiv preprint arXiv:2012.15460* (2020).
- [39] Pavel Tokmakov et al. “Learning to Track with Object Permanence”. In: *ICCV*. 2021.
- [40] James P Underwood et al. “Lidar-based tree recognition and platform localization in orchards”. In: *Journal of Field Robotics* 32.8 (2015), pp. 1056–1074.
- [41] Zhongdao Wang et al. *Towards Real-Time Multi-Object Tracking*. 2020. arXiv: 1909.12605 [cs.CV].
- [42] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. “Simple online and realtime tracking with a deep association metric”. In: *2017 IEEE international conference on image processing (ICIP)*. IEEE, 2017, pp. 3645–3649.
- [43] Yifu Zhang et al. “Bytetrack: Multi-object tracking by associating every detection box”. In: *European Conference on Computer Vision*. Springer, 2022, pp. 1–21.
- [44] Yifu Zhang et al. “Fairmot: On the fairness of detection and re-identification in multiple object tracking”. In: *International Journal of Computer Vision* 129 (2021), pp. 3069–3087.

- [45] Guan Zhaoxin et al. “Design a robot system for tomato picking based on yolo v5”. In: *IFAC-PapersOnLine* 55.3 (2022), pp. 166–171.
- [46] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. “Tracking Objects as Points”. In: *ECCV* (2020).