IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 March 2023, accepted 25 March 2023, date of publication 29 March 2023, date of current version 20 April 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3263238

RESEARCH ARTICLE

Early Yield Estimation in Viticulture Based on Grapevine Inflorescence Detection and Counting in Videos

MUHAMMAD RIZWAN KHOKHER[1], (Member, IEEE), QIYU LIAO[1], ADAM L. SMITH[2], CHANGMING SUN[1], DONALD MACKENZIE[2], MARK R. THOMAS[2], DADONG WANG[1], (Senior Member, IEEE), AND EVERARD J. EDWARDS[2]

[1]CSIRO Data61, Marsfield, NSW 2122, Australia
[2]CSIRO Agriculture and Food, Waite Campus, Urrbrae, SA 5064, Australia

Corresponding author: Muhammad Rizwan Khokher (rizwan.khokher@data61.csiro.au)

This work was supported by the Department of Agriculture, Fisheries and Forestry, through their Rural Research and Development for Profit Scheme and Wine Australia.

ABSTRACT In viticulture, yield estimation is a key activity, which is important throughout the wine industry value chain. The earlier that an accurate yield estimation can be made the greater its value, increasing management options for grape growers and commercial options for winemakers. For the yield estimate based on in-field measurements at scale, the number of inflorescences emerging after bud-burst offers the earliest practical signal, allowing a yield potential to be determined months before harvest. This paper presents an approach to automatically count the inflorescence number at the phenological stage E-L 12 using RGB video data and demonstrates its use for estimating yield. A dataset consisting of RGB videos was collected shortly after bud-burst from multiple vineyards, in conjunction with hand counts to produce a manual ground-truth for the inflorescence counting task. The video frames were annotated using bounding-boxes around the inflorescences to produce a digital ground-truth. A deep learning architecture was developed to learn features from the video frames during training and detect the inflorescences at the later inference stage. The detection results were fed to a tracking pipeline built using computer vision and deep learning techniques to generate numbers of inflorescences present in test videos. The visual and quantitative results are presented and evaluated for the inflorescence detection and counting tasks. The developed inflorescence detector achieves an average precision of 80.00%, a recall of 83.92%, and an F1-score of 80.48%, through a five-fold cross-validation on the annotated dataset. For the test videos, the developed automatic inflorescence counting model reports an absolute error of 11.03 inflorescences per panel, a normalized mean absolute error of 10.80%, and an R^2 of 0.86, when the predicted per-panel counts were compared to the corresponding manual ground-truth. Based on the counting results, we estimate an early yield that is within 4% to 11% error when compared to the actual yield after harvest. Based on these results and a separate analysis of the relationship between hand counts of inflorescences and harvest yields in three vineyards over three growing seasons, we conclude that computer vision and machine learning based methods have the potential to provide early yield estimation in viticulture with a commercially viable accuracy.

INDEX TERMS Grapevine, inflorescence detection, inflorescence tracking, early yield estimation, viticulture, computer vision, deep learning.

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

I. INTRODUCTION

Yield estimation is of increasing importance in agricultural industries as they become more vertically integrated. All

aspects of logistical planning can benefit, from crop insurance for growers to the delivery to processors or wholesalers, to the marketing of products. In the wine industry, where fruit quality can drive a ten-fold difference in the value of the crop for the same genetics, yield and fruit composition are commonly assumed to be linked, so vital in-season management decisions can also depend on accurate yield estimation at an early stage. Grapevine yield (per vine) is composed of three factors, bunch number, berry number per bunch, and mean berry weight. Bunch number is considered to be the largest single driver of season-to-season yield variation in Australian viticulture [1] and can potentially be estimated at a much earlier stage than the other factors [2], [3], [4]. The maximum number of bunches is limited to the number of inflorescences a vine produces, which in most production systems emerge and develop only at a certain phenological stage, during spring growth. Where no human intervention occurs and there are no adverse climatic events, such as frost, hail, or severe drought, the vast majority of inflorescences are retained and will form grape bunches. Inflorescence primordia are laid down during bud formation in the previous season, thus over-wintering buds can be dissected and the primordia counted, providing the earliest possible measure of potential fruitfulness in the subsequent growing season. However, this is a time-consuming and destructive process, expensive to undertake, does not determine the actual number of inflorescences that emerge at bud-burst, and cannot realistically be developed into an on-the-go measurement system. Consequently, the earliest estimate of the potential number of bunches that could be assessed at scale is a count of the number of emerged inflorescences shortly after bud-burst.

A. RELATED WORK

Current methods of yield estimation, whether bud dissection or in-field counts and weights during the season, are labour-intensive and rely on a sample density adequate to fully represent the variation across a vineyard block for accuracy, something that is very rarely achieved. Utilizing a non-destructive, on-the-go, method of yield estimation would not only overcome the problems associated with achieving a statistically valid sample density but also potentially provide a high-resolution spatial map of yield, which could be utilized by the vineyard manager using precision agriculture principles to manage the crop more efficiently and profitably ([5] and references therein).

Recent advances in computing power, computer vision, and robotics have provided the potential to provide just such an on-the-go system. The combination of image processing, computer vision, and machine learning techniques are becoming widely available and are already being used as management tools for agriculture [6]. In viticulture, these techniques have already been explored for some aspects of yield estimation [2]. Much of these have been to estimate bunch size or berry number per bunch close to harvest [2], [7], [8], but some attempts have been made at early inflorescence stages when flowers are separate and about to open. To date, these have aimed to measure the number of flowers within inflorescences, providing a potential maximum berry number. A variety of imaging, both indoor and outdoor in a controlled or uncontrolled environment with natural or artificial backgrounds through a destructive or non-destructive manner have been used. A range of image analysis, computer vision, and deep learning techniques have been applied for flower counting within inflorescence images [9], [10], [11], [12], flower counting within segmented inflorescences [13], [14], [15], [16], opened and unopened flower classification within inflorescence images [17], and spikelet detection within detected inflorescences for thinning analysis [18]. A summary of these approaches is given in Table 1. All of these methods used only static images and none of them worked with video sequences. Only one of the methods in [15] provided a yield estimation through a relationship between predicted number of flowers versus actual yield per vine. Furthermore, all of these methods work with images of inflorescences at or after E-L 15 stage [19] when flowers are usually separate and quite visible just before flowering.

If used for early yield estimation, the above techniques based on individual flower count per inflorescence would typically rely on a manual estimate of the total number of inflorescences and an empirical relationship between the flower number and berries set for a given plot. Given the aforementioned relationship between the bunch number and inter-seasonal variation and the requirement to predict yield at the earliest possible time (E-L 12 stage [19]), we aimed to develop a new technique that would provide on-the-go counts of grapevine inflorescences from videos, which could be combined with historical bunch weight information to provide an early estimate of yield at an earlier E-L 12 stage, or simply compared with inflorescence counts obtained in previous seasons as an index of yield potential.

B. CONTRIBUTIONS

There are a few problems to overcome while developing such an approach based on inflorescence counts. Firstly, there is no public dataset available for inflorescence detection and counting tasks in videos. A few datasets can be found but only for the bunch and berry detection in images, at the veraison or harvest stages. To this end, a new video dataset needs to be collected and labelled. Secondly, efficient detection of inflorescences is required in videos. This is because localization and detection of inflorescences is a very challenging task due to complex backgrounds with leaves, stems, and occlusions. Thirdly, the inflorescence counting needs to be performed as soon as inflorescences are visible (E-L 12 stage [19]) with less occlusion by leaves, which is before the individual flowers are separated (E-L 15 stage [19]).

To solve the aforementioned problems, in this work, the major contributions include:

1. A new dataset was collected in the form of RGB videos at the inflorescence E-L 12 stage. The videos were taken in a natural environment without any disturbance or

TABLE 1. Related methods using image analysis, computer vision, and deep learning techniques at grapevine inflorescence stages.

Method	Year	Environment	Approach	Techniques
Diago et al. [9]	2014	Controlled, indoor, plain background, destructive	Flower counting within inflorescences	Image analysis
Aquino et al. [10]	2015	Controlled, indoor, plain background, destructive	Flower counting within inflorescences	Image analysis
Aquino et al. [13]	2015	Uncontrolled, outdoor, natural background, non-destructive	Flower counting within segmented inflorescences	Image/vision analysis
Millan et al. [11]	2017	Controlled, indoor, plain background, destructive	Flower counting within inflorescences	Image analysis
Liu et al. [12]	2018	Controlled, outdoor, plain background, non-destructive	Flower counting within inflorescences	Image analysis
Rudolph et al. [14]	2018	Uncontrolled, outdoor, natural background, non-destructive	Flower counting within segmented inflorescences	CNN and image analysis
Pahalawatta et al. [17]	2020	Controlled, outdoor, plain background, non-destructive	Flower classification within inflorescences if opened/unopened	CNN and image analysis
Palacios et al. [15]	2020	Uncontrolled, outdoor, natural background, non-destructive	Flower counting within segmented inflorescences	CNN and image analysis
Du et al. [18]	2022	Uncontrolled, outdoor, natural background, non-destructive	Spikelet detection within inflorescences for thinning analysis	CNN and image analysis
Rahim et al. [16]	2022	Uncontrolled, outdoor, natural background, non-destructive	Flower counting within segmented inflorescences	CNN and image analysis

customization of the background. The video frames were labelled manually to provide ground-truth bounding-boxes for inflorescences as per the standards of object detection in computer vision. A part of this dataset was made public for other researchers to work with.

2. To efficiently detect inflorescences in videos, an inflorescence detector was developed based on convolutional neural network (CNN) architectures. The detector was trained using the collected inflorescence dataset which can detect inflorescences, in the wild and in different weather, lighting, and background conditions.
3. A new tracking pipeline was developed using computer vision and deep learning approaches to automatically track the detected inflorescences and count them in videos. Both the detector and the tracker were evaluated rigorously using the ground-truth data produced during and after the data collection.
4. The developed inflorescence detection model and the tracker were tested for a selected number of videos/rows of two different grapevine varieties in different weather and lighting conditions. The panel-wise and row-wise inflorescence counting results were evaluated using ground-truth counts. In the end, the counts were used to produce an early yield estimate and compared with the actual yield for that season.

A block diagram of the proposed framework for the early yield estimation based on inflorescence detection and counting in videos is given in Fig. 1. Every component of the framework is discussed in detail in the following sections.

II. GRAPEVINE INFLORESCENCE DETECTION AND COUNTING

In this section, firstly, we present the developed inflorescence detector based on a CNN for the efficient detection of inflorescences in RGB videos. Secondly, our K-shortest paths (KSP) and deep learning-based inflorescence tracker is described to track and count the inflorescences in videos. Thirdly, we describe the imaging and video datasets collected and labelled for the inflorescence detection and counting tasks, followed by the discussion on the experimental setups, pre and post-processing of data, hyper-parameters of our proposed machine learning models, and evaluation protocols.

A. THE INFLORESCENCE DETECTOR

Accurate detection of inflorescences is a critical component of the vision-based early yield estimation system in viticulture. The task is to detect inflorescences in an outdoor and contactless scenario where the videos have naturally complex backgrounds. The detection accuracy is affected by occlusions, complex backgrounds, and multiple targets. As in our case, there can be many inflorescences present in a single video frame, any number of which could be occluded by other inflorescences and leaves. Furthermore, there can be other rows and trees in the background of inflorescences. These challenges make the inflorescence detection problem difficult. Recently, deep learning approaches have been widely and successfully used for various object detection tasks [21], [22]. A similar methodology can be adapted for the inflorescence detection task.

Some of the widely used CNN architectures include VGG [23], GoogLeNet [24], ResNeXt [25], HRNet [26], and Swin-Transformer [27]. There have been many object detectors designed by building upon the above architectures. Such object detectors can be divided into three categories: one-stage, two-stage, and multi-stage detectors. One-stage detectors including OverFeat [28], you-only-look-once (YOLO) [29], and RetinaNet [20], are fast for near real-time detection but they compromise on localization ability. Whereas two-stage detectors including Faster-RCNN [30], feature pyramid networks (FPN) [31], and EfficientNet [32], and multi-stage detectors including Cascade-RCNN [33] and hybrid-task cascade (HTC) [34], have strong localization ability but with added complexity. In this work, a deep learning-based object detection framework was developed based on a combination of ResNeXt, FPN, and RetinaNet to localize and detect inflorescences in videos.

The architecture of the inflorescence detector is shown in Fig. 2. The deep features are extracted using ResNeXt acting as a backbone network that uses group convolutions to reduce the number of parameters and increase accuracy. A ResNeXt architecture with 101 layers is used with 64 group convolutions. The extracted features at different layers can be used to build a pyramid structure to extract multi-scale features. For this purpose, a feature pyramid network is used to exploit the natural pyramid structure of the backbone network. The multi-scale features from the FPN network are then used by RetinaNet to produce the bounding-box detections and

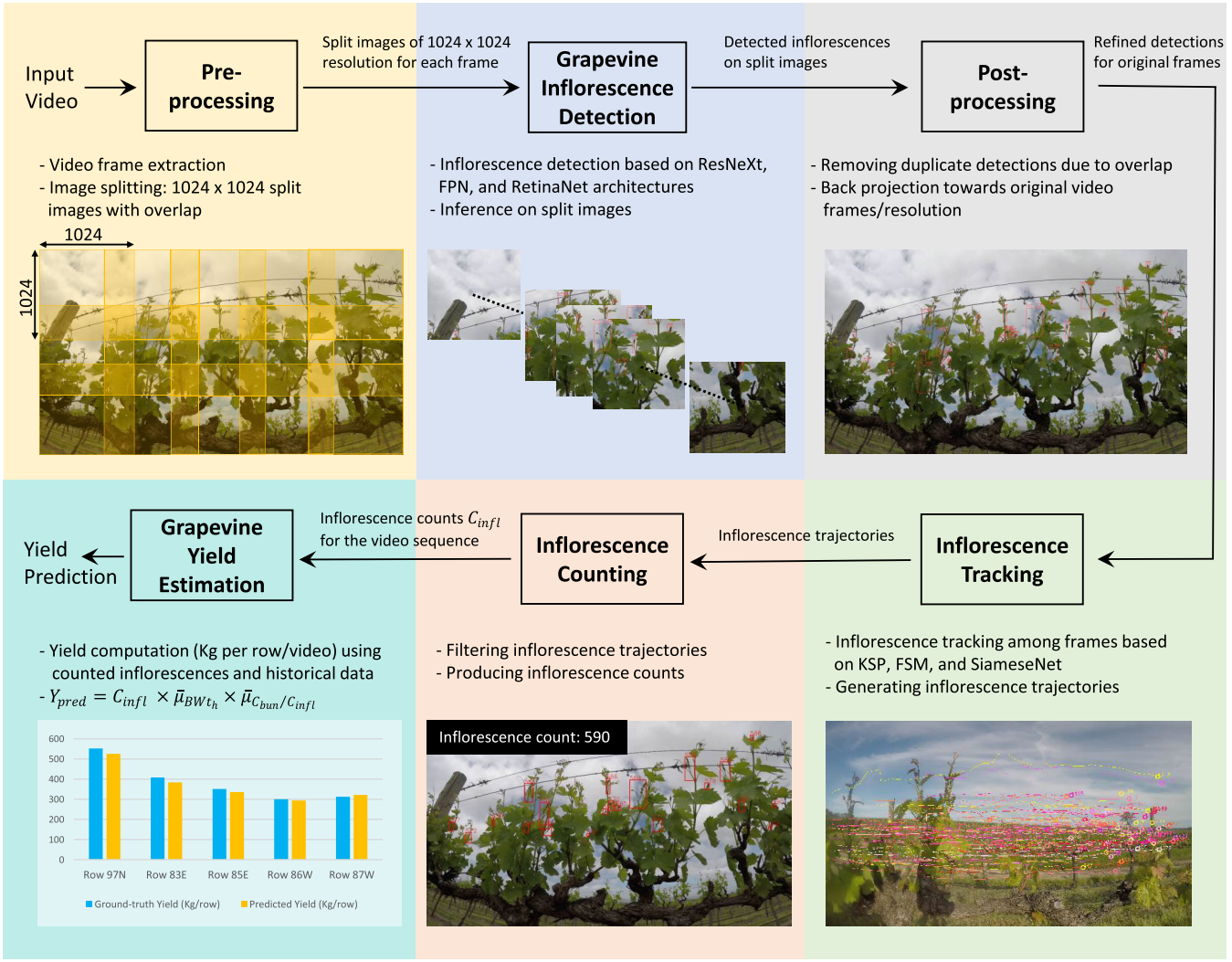


FIGURE 1. Block diagram of the proposed early yield estimation framework based on inflorescence detection, tracking, and counting in videos.

classes associated with those detections. RetinaNet makes use of anchor boxes and focal loss to produce the detections and optimise the network weights, respectively. There are two subnets: Class Subnet and Box Subnet. A Class Subnet is a fully-connected network to predict the probability of an object present at a spatial location; whereas a Box Subnet is a fully-connected network to output locations of a bounding-box around a classified object.

B. TRACKING BASED INFLORESCENCE COUNTING

The inflorescence detector is run on videos captured on-the-go using cameras mounted on a *Kubota* vehicle. To count the inflorescences in the videos, individual inflorescences need to be tracked in subsequent frames. During tracking, there are different problems that can arise. For example, occlusions by vines, leaves, and other inflorescences. Due to the occlusions, an inflorescence may disappear at a location in a video frame and re-appear at another location of a different frame after some time. Appearance-based strategies to match the bounding-boxes for an object in different frames may not work in this case, as all the target objects (inflorescences) have similar shapes and appearances. In addition, the depth of field varies for the inflorescences in the middle of the video frame than the ones on either side. This causes irregular motion patterns, and the camera motion estimation becomes difficult to predict. Furthermore, there can be missing detections in the video frames along with false positives which makes the tracking problem more complex.

To build the inflorescence detector based on tracking, KSP [35] is used to track the inflorescences as a series of disconnected trajectories, called tracklets. A finite state machine (FSM) is then designed to connect the tracklets into continuous trajectories which result in inflorescence counts, as described in Section II-B2.

1) GRAPH CONSTRUCTION USING K-SHORTEST PATHS

Given the i-th detection in the j-th frame as node N_j^i, which has five parameters: $P_i^j = (x_1, x_2, y_1, y_2, c)$. Here,

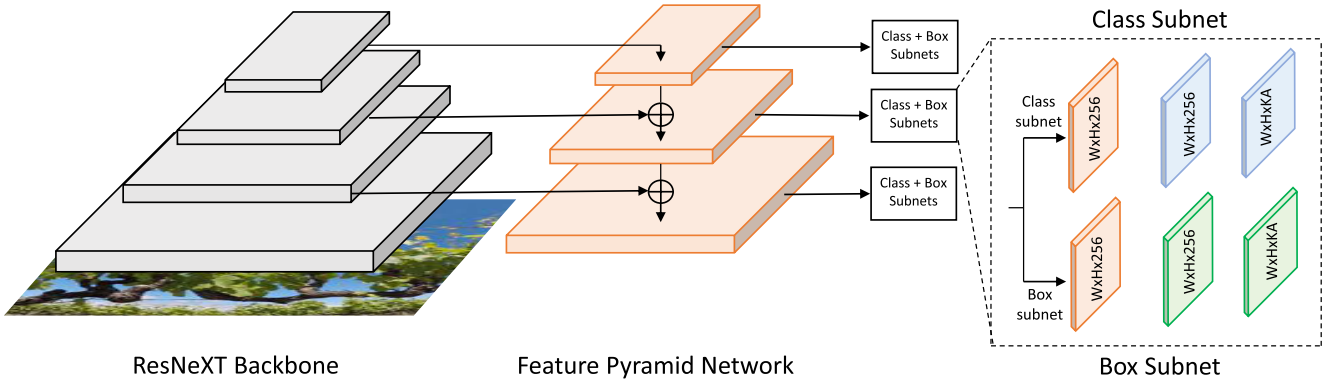


FIGURE 2. The inflorescence detector based on ResNeXt, FPN, and RetinaNet adapted from [20].

(x_1, y_1) and (x_2, y_2) represent the top-left and bottom-right corners of a detected bounding-box, whereas c is the confidence of the detection. Besides the nodes of the detections, following [35], a weighted graph G is constructed. Firstly, the source node N_{source} is linked to every detection node in every frame. The source edge cost C_{source} is defined as $C_{source} = -\log\frac{c}{1-c}$, where c is the confidence score of the detection. Then all the detection nodes are linked in every frame to the sinking node N_{sink} without setting any punishment in the sinking edges ($C_{sink} = 0$).

Each detection node is linked in frame j to every detection node in frame $j+1$. The edge cost of the n-th detection node in the j-th frame to the m-th detection node in the $(j+1)$-th frame $C_{n,j}^{m,j+1}$ is defined as:

$$C_{n,j}^{m,j+1} = -\log\frac{c_{m,j+1}}{1-c_{m,j+1}} + \mu \times \min(\Delta x_1, \Delta x_2, \Delta y_1, \Delta y_2), \tag{1}$$

where $c_{m,j+1}$ is the confidence score of the m-th detection in the $(j+1)$-th frame. $(\Delta x_1, \Delta x_2, \Delta y_1, \Delta y_2)$ is the location change of the two detections between two frames, and μ is a combination factor. The weighted graph is solved by adapting the approach in [35].

2) SiameseNet BASED INFLORESCENCE TRACKING

Missing detections in some video frames due to occlusions can cause multiple trajectories for inflorescences resulting in duplicate counting. To fix this problem, we adapt SiameseNet (SiamFC) [36] to match the detected inflorescences in surrounding regions and locate them in the subsequent frames. The inflorescence tracker based on SiamFC is shown in Fig. 3. Based on [36] and following the idea of tracking bounding-boxes and masks in [37] and [38], we use the depth-wise cross-correlation between the two feature streams and use the Response of a candidate Window (RoW) to localise the bounding-boxes with certain confidence scores in the subsequent frames. A pre-trained model of [37] on ImageNet-VID [39] is used and validated on our video data.

The proposed FSM to predict the state of each trajectory is demonstrated in Fig. 4. There are four possible states for a trajectory: DETECTED, TRACKED, BLOCKED, and END. DETECTED is the initial state when a new trajectory is created. TRACKED denotes that the trajectory can be associated with detection or tracked by SiamFC in the new frame. BLOCKED denotes that the trajectory cannot find an association and cannot be tracked by SiamFC in the new frame. END means the trajectory is blocked for too long or moves out of view. We solve the FSM using the Hungarian algorithm [40] and SiamFC. The detailed state transferring conditions and parameters are described in Section IV-B.

III. GRAPEVINE INFLORESCENCE DATASET

A new video dataset was collected at the inflorescence stage and curated for the inflorescence detection and counting tasks. As there is no public dataset available for these tasks, a part of the dataset has been made publicly available to enable other researchers to reuse and analyze the data. A set of annotated RGB images for the inflorescence detection task is available at https://doi.org/10.25919/5de4546aeacce.

A. VIDEO DATA ACQUISITION

The RGB videos were captured at the inflorescence stage at two vineyards, one at Woodside and another at McLaren Vale, both in South Australia. The grape varieties include *Tannat* and *Shiraz*, with vertical shoot position (VSP) and 'sprawl' canopy management, at Woodside and McLaren Vale, respectively. To capture the videos, GoPro Hero 5 and 7 cameras were used. The cameras were mounted on a Kubota ground vehicle driving at speeds of 3 to 4 km/h, at a distance of nearly one metre away from the canopy, as shown in Fig. 5(a). The videos were captured from three different camera views of the canopy: top, middle, and bottom, as shown in Fig. 5(b). The distance between the cameras was 30cm. The cameras were set to 30°, 45°, 60° angles for the top, middle, and bottom views, respectively. The video capturing was performed in an ''on-the-go'' manner without customization or interaction with vines. This was to approximate an acquisition method that would be easily amenable to commercial use. There was no customization performed on the vines and the data collection was carried out in a contact-less manner.

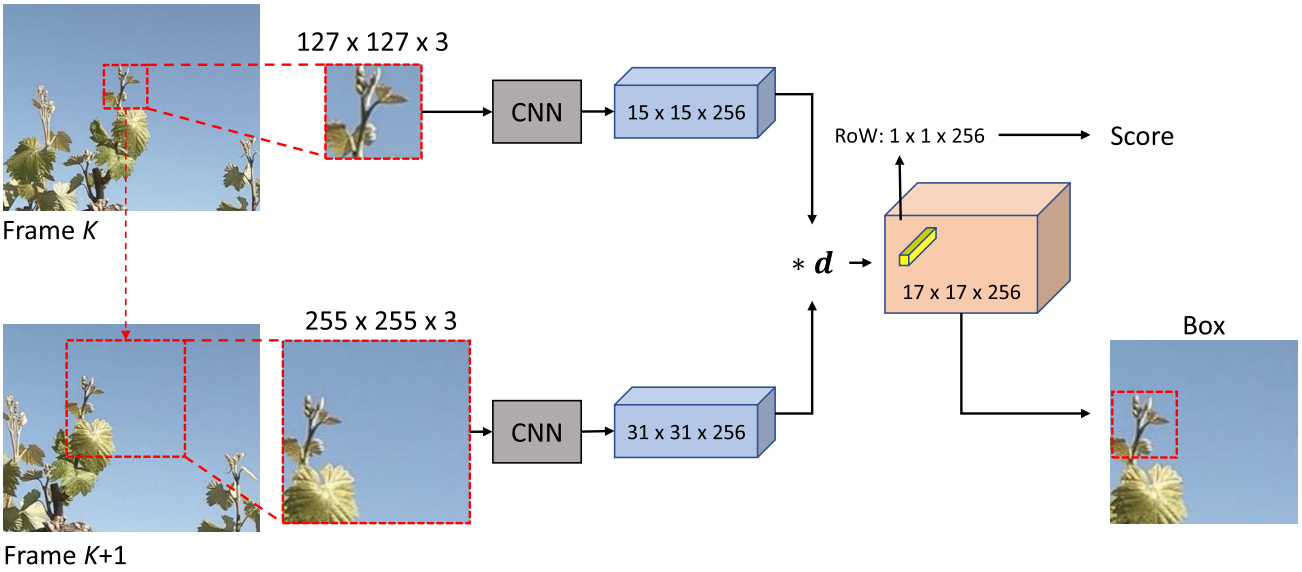


FIGURE 3. Block diagram of the inflorescence tracker based on SiameseNet [36].

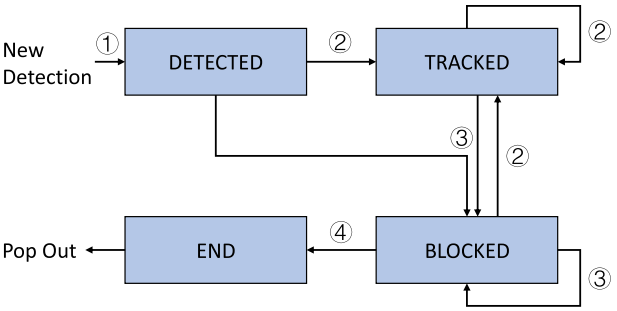


FIGURE 4. The FSM with different states and conditions of the transitions.

The videos were captured at resolutions 4,000 × 3,000 and 3,840 × 2,160 with a frame-rate of 30 frames per second using auto exposure. The videos were captured in conditions with all variations likely to be encountered in a commercial setting, as shown in Fig. 6. The variations in the background include the sun, clouds, trees, and other rows, whereas the variations in illumination include low and bright light during the cloudy and sunny weather, respectively. These variations make the detection problem challenging. Furthermore, inflorescences become partially visible due to occlusions by leaves, stems, or other inflorescences. This makes the problem more complex.

B. PREPARING THE GROUND-TRUTH

To obtain the digital ground-truth, inflorescences were labelled manually using bounding-boxes in the extracted video frames. The images were labelled by experts in the field of viticulture. An open-source software called QuPath [41] was used to annotate videos by drawing bounding boxes around inflorescences. The bounding-box corners as (x, y) coordinates were extracted for the labelled video frames. A total of 800 frames were labelled which include variations in background and illuminations, as shown in Fig. 6.

(a) Kubota vehicle (b) Cameras mounted with different views

FIGURE 5. (a) Kubota vehicle used for video data collection. (b) Three different cameras are mounted on the vehicle.

A sample video frame with its close-up image, labelled using QuPath, is shown in Fig. 7. The ground-truth inflorescences are shown in black coloured bounding-boxes. The bounding-boxes enclose the inflorescences from the bottom (attachment point) to top, also a multi-branch inflorescence was considered as one.

For the counting task, five rows of inflorescences were selected for testing our inflorescence detector, which were not used for training the detector. The videos for the five test rows exhibit the variations discussed previously. To provide the ground-truth for the counting task, the inflorescences were counted manually in the field at the time of data collection. For these videos, panel-wise counts were produced as manual ground-truth for counting. Four of the rows were counted from both sides to counter any missing counts. The details of the videos for the five test rows are given in Table 2.

FIGURE 6. Sample video frames from the collected dataset. The black bounding-boxes represent the ground-truth inflorescences. The videos contain backgrounds with (a) sun, (b) other rows, (c) clouds, and (d) trees, and different lighting conditions (a, c) bad lighting (b, d), good lighting.

The viticultural practice at the Woodside vineyard includes significant bunch thinning, potentially reducing the link between inflorescence number and bunch number at harvest. Further, per panel harvest data were not able to be collected from the McLaren Vale rows. To support the counting data collected in conjunction with the video data acquisition, hand counting of inflorescences, and bunches at harvest and measurement of harvest weight were conducted over three growing seasons on each panel for five additional rows of vines at McLaren Vale.

IV. EXPERIMENTAL SETUP

The experiments were conducted using a Tesla P100-SXM2 GPU with 16 GB of memory. The implementation of the inflorescence detector was adapted from the MMDetection toolbox from the OpenMMLab [42]. This toolbox makes use of libraries including Python, PyTorch, and OpenCV. For the quantitative evaluation, a five-fold cross-validation approach was adopted. That is for each fold, the labelled video frames (as described in Section III-B) were split into 80% for training and 20% for testing of the inflorescence detector. Horizontal flipping of the training images was performed as data augmentation to increase the training data. For training, transfer learning was performed. That is, the deep learning network was initialized using a pre-trained network (trained on a public object detection dataset), and then fine-tuned on our inflorescence detection dataset.

The inflorescence detection results were obtained for the five test videos (as described in Section III-B). The inflorescence detector was run at every frame of the test videos and the spatial locations of the detected bounding-boxes with their confidence scores were stored in JSON files. The detection results were then used by the inflorescence tracker to produce panel-wise counts for the test videos. To compare the panel-wise counting of the tracker with the ground-truth, a key frame was selected manually for the first panel when the post (a wooden pole separating two panels) is at the centre of the view. Then the key frames of the upcoming panels were estimated based on the vehicle speed.

A. DATA PRE/POST-PROCESSING

The video frames are of high resolution. To fit the video frames of $4{,}000 \times 3{,}000$ and $3{,}840 \times 2{,}160$ into the GPU, the frames were split into several smaller images of $1{,}024 \times 1{,}024$ pixels with some overlap. The amount of overlap was calculated based on the size of the largest ground-truth bounding-box present in the corresponding frame. The bounding-box locations were then re-calculated for the smaller images which were then used for training and testing of the inflorescence detector. During testing, the

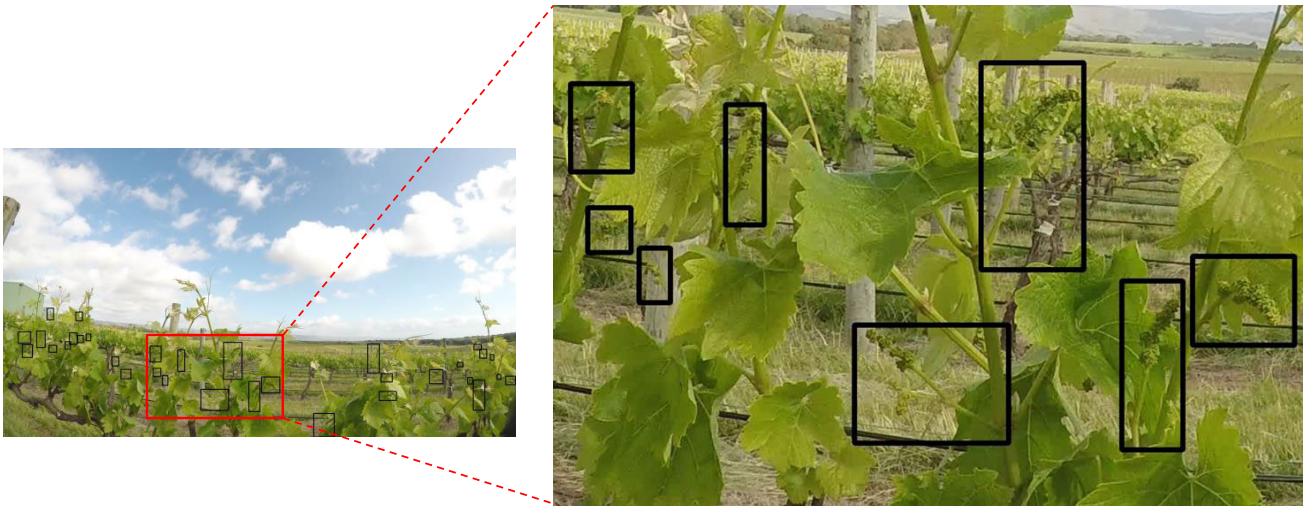


FIGURE 7. A sample video frame labelled using QuPath [41]. The ground-truth bounding-boxes for the inflorescences are shown in black colour.

TABLE 2. Test videos/rows of inflorescences for the counting task.

Row	No. of frames	No. of panels	Double-sided counts
R83-east	8,330	29	Yes
R85-east	7,785	29	Yes
R86-west	7,725	28	Yes
R87-west	8,741	28	Yes
R97-north	13,286	39	No

bounding-box locations of the detected inflorescences on the smaller images were projected back to the original resolution for the test frame. There may be duplicated detections due to the overlap which were removed during the back-projection of the detected bounding-boxes.

B. PARAMETER SETTINGS

For each fold of the five-fold cross-validation, the batch size was set to 1 and the learning rate was set to 0.001 during the training. After pre-processing, 640 frames for training were split into 12,680 smaller images. The inflorescence detector network was fine-tuned on these images for 15 epochs after which there was no significant reduction in training loss. The trained network with the best results was selected as the final detection model. During the testing of the inflorescence detector, intersection-over-union (IOU) was calculated for the predicted bounding-boxes. For a predicted bounding-box with area A and the corresponding ground-truth with area B, the IOU is defined as $(A \cap B)/(A \cup B)$. The IOU threshold was set to 0.5 which is standard across different object detection platforms. This means if a detected bounding-box had a 50% overlap with the corresponding ground-truth, it was considered a true-positive (TP), otherwise, it was considered a false-positive (FP). If there were multiple detections on the same ground-truth, then one of them was considered as a TP and the others as FPs. The minimum confidence value for a bounding-box detection was set to 0.3 for accepting it as a valid detection. This means that the predicted bounding-boxes with at-least 30% confidence were selected, and the rest were discarded. This threshold was tuned through a grid search during the experiments.

During inflorescence tracking, the following set of rules and parameters are used by the FSM in Fig. 4:

i. A new trajectory is created and enters the state of DETECTED when a potential inflorescence is detected and not associated with any previous trajectories. The confidence score threshold is set to 0.995. We define the distance of a detection to a trajectory as the distance between the detection and the predicted location of the trajectory.
ii. When a new detection can be associated with a previous trajectory, the location is updated and the new location is estimated in the next frame. The average speed of the previous five frames is used to predict the location in the next frame. If there no detection can be associated with the trajectory in the new frame, we crop the predicted region and use a pre-trained SiamFC to localize the trajectory in the new frame, as described in Section II-B2.
iii. When a trajectory cannot find any detection in the current frame and the confidence score is lower than a threshold, which is set to 0.5 based on experiments, the trajectory is assumed to be occluded and the previously saved information is used to estimate the motion pattern. In the meantime, the location prediction keeps updating for new frames.
iv. If a trajectory is BLOCKED for more than five frames or the estimated location is out of view, we pop the trajectory out of the tracking list.

C. EVALUATION METRICS

To quantitatively evaluate the detection results, precision, recall, and F1-score are calculated as $p = \text{TP}/(\text{TP} + \text{FP})$, $r = \text{TP}/(\text{TP}+\text{FN})$, and $\text{F1-score} = 2\times\left(\frac{p \times r}{p + r}\right)$, respectively, FN refers to false-negative. Precision gives a percentage that

shows how accurately a model predicts and recall gives a percentage that shows how many actual targets are detected out of the total targets. At each detection, a pair of precision and recall values is obtained to draw a curve called the recall-precision curve (RPC). From this curve, average precision (AP) is calculated which provides a quantitative score that shows how good the detection model is. The AP is calculated by finding the area under the RPC by interpolating over all levels of recall as:

$$\mathrm{AP} = \sum (r_{n+1} - r_n) p_{interp}(r_{n+1}), \tag{2}$$

$$p_{interp}(r_{n+1}) = \max_{\tilde{r} \geq r_{n+1}} p(\tilde{r}), \tag{3}$$

where $n = 0$ to all, r_n represents the nth recall value, $p_{interp}(r_{n+1})$ represents the interpolated precision at recall level r_{n+1}.

To quantitatively evaluate the inflorescence counting results, mean absolute error (MAE), root mean square error (RMSE), and the coefficient of determination R^2 are used. MAE and RMSE are given as:

$$\mathrm{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|, \tag{4}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2}, \tag{5}$$

where y_i represents the ground-truth value and $\hat{y}_i$ represents the predicted value. N represents the total number of samples. R^2 represents the proportion of the variation in the dependent variable that is predictable from the independent variable, and it can be calculated as:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}, \tag{6}$$

where i ranges from 1 to the total number of y and $\hat{y}$ pairs, $\bar{y}$ represents the mean of ground-truth values y_i.

V. RESULTS AND DISCUSSION

In this section, the inflorescence detector's training process followed by some visual and quantitative results is discussed. The tracking and counting results for the test dataset are also analysed. The overall and panel-wise inflorescence counts are presented and discussed. Based on the automatic counting results for the test videos, an estimate of the yield is provided and compared with the actual yield after harvest. The evaluation metrics described in Section IV-C are calculated to support the above-mentioned analysis.

A. INFLORESCENCE DETECTION RESULTS

1) TRAINING THE DETECTOR

There were four different deep learning based detectors including OverFeat, YOLOv5, Cascaded-RCNN, and RetinaNet, explored to develop our inflorescence detector. The inflorescences can be very small and hard to detect in the video frames, the best detection results were obtained using RetinaNet, chosen for our final model development of the inflorescence detector.

The inflorescence detector was trained on the training dataset with the hyper-parameters described in Section IV-B. An epoch means going through the whole training data once. For batch size 1, an iteration represents processing one image at a time. This means each epoch contains 12,680 images represented by 12,680 iterations (Fig. 8). To complete the training process, the total computational time taken by a single GPU of 16 GB memory was around 68 hours.

The behaviours of different losses during the training process are discussed. The graphs for class loss, bounding-box loss, and overall training loss are shown in Fig. 8. The class loss starts decreasing from 1.08 and quickly reaches 0.27 just after 5,100 iterations. The training was stopped and resumed at 101,440 (8 epochs) iterations which causes a quick drop in the loss to 0.034. This is because the training parameters were reset. The minimum loss obtained is 0.004. Similarly, the bounding-box loss has a starting point of 0.45 which reaches swiftly 0.186 after 5,900 iterations. After 101,440 iterations, a further decline is observed that is 0.043 and a minimum loss of 0.014 is achieved. For the overall training loss, a significant reduction in loss is seen from 1.53 to 0.45 in the first 5,100 iterations. A minimum overall training loss of 0.018 is achieved. Multiple models were obtained after training for different numbers of epochs and the above graphs helped in selecting the appropriate model which could provide the optimal results.

2) VISUAL RESULTS AND ANALYSIS

In this section, some visual results are presented and discussed. The experiment settings described in Section IV were used during the training. The detected inflorescences are shown in Fig. 9 and Fig. 10 for some of the test video frames in different lighting and background conditions, respectively. The ground-truth inflorescences are represented by the bounding-boxes in blue colour, whereas the detected inflorescences are represented by red coloured bounding-boxes. The percentages above the bounding-boxes represent the confidence scores of the predictions.

The detector can efficiently detect the inflorescences which are either completely visible with clear texture as shown in Fig. 9(b) or partially occluded by leaves, stems, or other inflorescences as illustrated in Fig. 9(a). Most of the detected inflorescences have confidence scores of more than 90% which is promising. The results show that the detector can work well in different lighting conditions, e.g., sunny and cloudy weather (Fig. 9). In addition, the inflorescence detector is robust for different backgrounds with other rows (Fig. 10(a)) and trees (Fig. 10(b)). Although some of the inflorescences were not labelled as ground-truth due to heavy occlusions, the detector can still detect those inflorescences. This demonstrates that the detector can easily detect the texture presented in the inflorescences with heavy occlusions that human expert annotators cannot always easily identify.

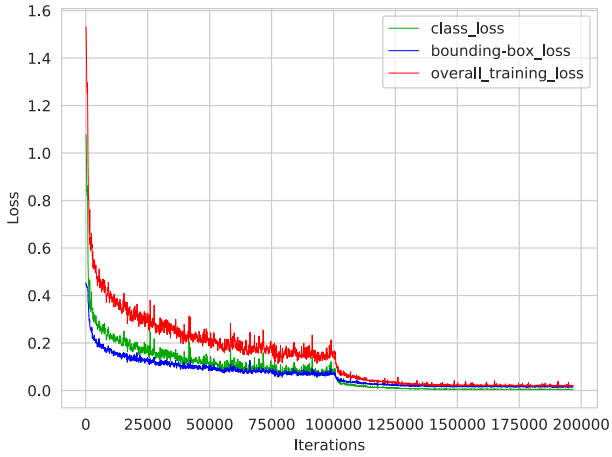


FIGURE 8. Graphs of different losses during training. The losses include class loss, bounding-box loss, and overall training loss.

TABLE 3. Different evaluation metrics obtained on the test dataset by the intermediate models trained for different numbers of epochs.

Epoch	No. of Detections	Recall (%)	AP (%)	F1-score (%)
1	TPs: 2,184, FPs: 2,341	79.5	86.5	82.8
2	TPs: 2,178, FPs: 974	79.3	70.7	74.8
3	TPs: 2,371, FPs: 1,668	86.3	75.7	86.7
4	TPs: 2,288, FPs: 1,715	83.3	73.6	78.2
5	TPs: 2,299, FPs: 1,031	83.7	77.1	80.3
6	TPs: 2,373, FPs: 1,319	86.4	79.2	82.6
7	TPs: 2,354, FPs: 1,798	85.7	76.4	80.8
8	TPs: 2,318, FPs: 2,271	84.4	73.7	78.7
9	TPs: 2,335, FPs: 745	85.0	80.1	82.5
10	TPs: 2,343, FPs: 753	85.3	80.4	82.8
11	TPs: 2,343, FPs: 556	85.3	81.1	83.2
12	TPs: 2,332, FPs: 721	84.9	80.2	82.5
13	TPs: 2,335, FPs: 723	85.0	80.2	82.5
14	TPs: 2,338, FPs: 734	85.1	80.2	82.6
15	TPs: 2,335, FPs: 730	85.0	80.2	82.5

3) QUANTITATIVE RESULTS AND ANALYSIS

The performance of the inflorescence detector is evaluated quantitatively using the metrics including average precision, recall, and F1-score. The experiment set-up is described in Section IV and the hyper-parameters for the inflorescence detector are detailed in Section IV-B. During inference, 160 test video frames were split into smaller images of size 1,024 × 1,024, resulting in 3,120 images for testing. The performance of the inflorescence detector trained for different numbers of epochs is analysed here. During the five-fold cross-validation, the inflorescence detector was trained for 15 epochs for each fold. The intermediate models for different numbers of epochs were obtained and tested on the test dataset.

For one of the five-folds, the quantitative results as the number of TPs and FPs, recall, and AP, are shown in Table 3. The corresponding RPCs of the intermediate models obtained on the test subset are shown in Fig. 11. From the RPCs, we can see that the curves for the first few epochs go downhill very quickly. For epochs 9 to 15, the RPCs look quite stable and similar, and this is where the training has converged. To further analyse the performance of the intermediate models, we can refer to Table 3. For epochs 9 to 15, the average precision and F1-score do not change much and remain within ranges of 80.1% to 81.1% and 82.5% to 83.2%, respectively. Similarly, the recall varies only a little from 84.9% to 85.3%. This can further be analysed by looking at the number of TPs and FPs. We need to minimise the number of FPs and maximise the number of TPs. Only the model trained for 11 epochs provides these numbers, i.e., FPs: 556 and TPs: 2,343, resulting in an AP of 81.1%, a recall of 85.3%, and an F1-score of 83.2%. This model was finalised for the current fold of the five folds. For the five-fold cross-validation, mean±standard error of the mean is given as, average precision: 80.00%±2.04%, recall: 83.92%±1.86%, and F1-score: 80.48%±1.48%, achieved by the inflorescence detector on the annotated dataset. The above number of FPs is affordable because for the next step in the overall pipeline (i.e., inflorescence counting) the inflorescence detection is performed at every frame in a test video and FP detections may not appear consistently in the subsequent frames. Such FP detections can be removed through the tracking process later during the counting of inflorescences.

B. INFLORESCENCE TRACKING AND COUNTING RESULTS

In this section, the performance of the developed inflorescence tracker and counter is evaluated on the test videos. Firstly, the process of tracking and counting inflorescences is analysed visually. Secondly, the average panel-wise and total row-wise, ground-truth and predicted counting results are presented for the test videos. Thirdly, the counting results are quantitatively evaluated by comparing them with the ground-truth. Lastly, the performance of the counter is analysed for different panel densities and lighting/background variations.

1) VISUAL RESULTS

The inflorescence detection results from the inflorescence detector were fed to the tracker for the five test videos/rows, as described in Section III-B. The experiment set-up and parameters for the tracking and counting models are described in Sections IV and IV-B, respectively. Some visual results for the tracking and counting of inflorescences are presented in Fig. 12 for different lighting conditions and backgrounds. The red bounding-boxes represent the tracked inflorescences and the numbers above the bounding-boxes represent the counts. The overall counts to a particular frame are given at the top left corner of the frame. We can see that the inflorescence counter performs quite well for all types of variations mentioned above.

2) QUANTITATIVE RESULTS

The ground-truth and predicted inflorescence counting results for the five test videos/rows: R83-east, R85-east, R86-west, R87-west, and R97-north, are presented here. Only videos with bottom camera views were used which

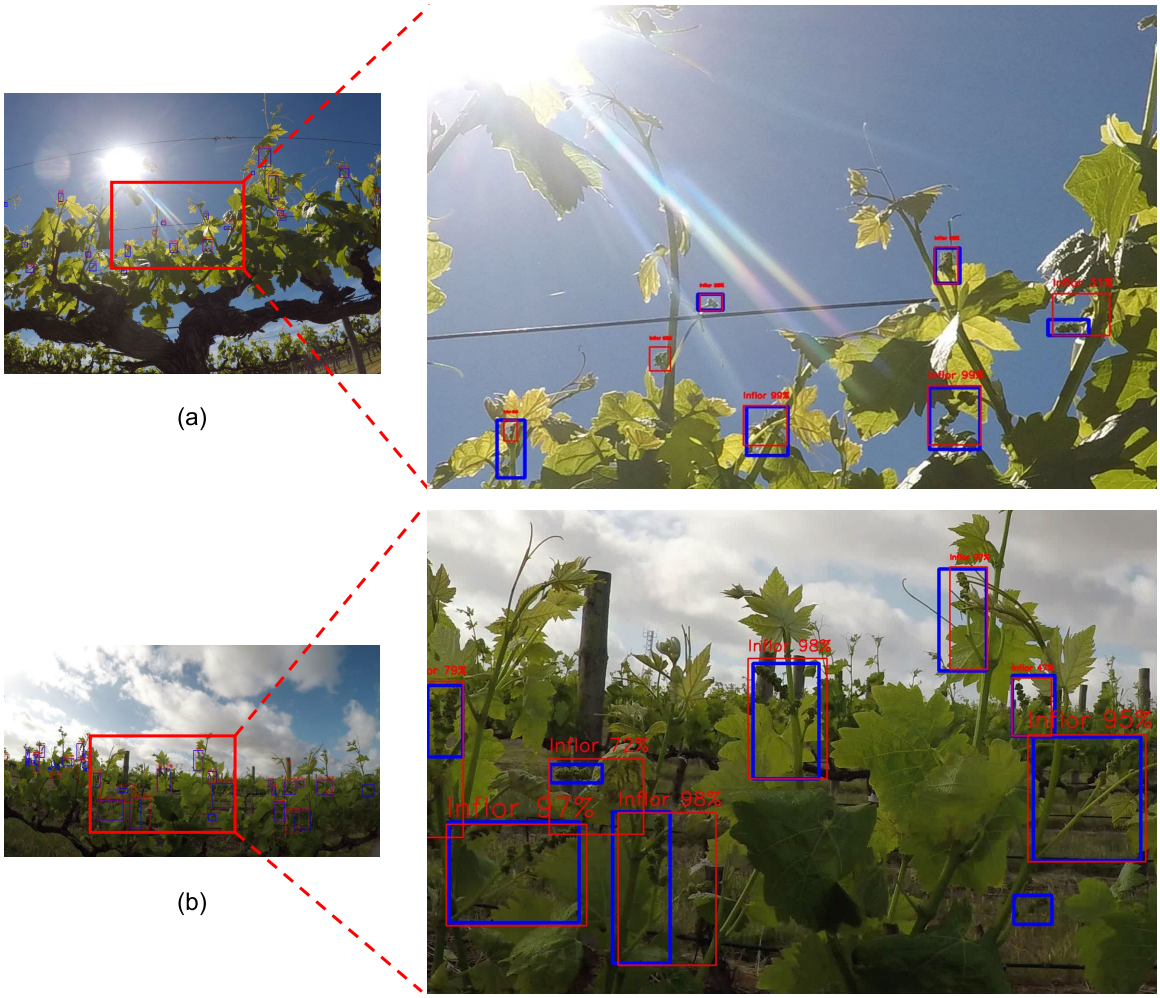


FIGURE 9. Inflorescence detection results on test video frames of different lighting variations. The ground-truth and detected bounding-boxes are represented in blue and red colours, respectively. The percentages above the bounding-boxes are the confidence scores. (a) A video frame with the sun in the background and high exposure. (b) A video frame in bad light with other rows in the background.

TABLE 4. Average panel-wise inflorescence counting results for the five test videos along with NMAE errors.

Row	Ground-truth mean counts (per panel)	Predicted mean counts (per panel)	NMAE
R83-east	90.62	93.76	3.46%
R85-east	86.76	82.41	5.00%
R86-west	80.04	75.32	5.88%
R87-west	83.46	78.57	5.85%
R97-north	151.60	143.46	5.34%

TABLE 5. Total row-wise inflorescence counting results for the five test videos along with NMAE errors.

Row	Ground-truth total counts (per row)	Predicted total counts (per row)	NMAE
R83-east	2,628	2,742	4.34%
R85-east	2,516	2,390	5.00%
R86-west	2,241	2,109	5.89%
R87-west	2,337	2,288	2.10%
R97-north	5,912	5,598	5.31%

showed less occlusions and complexity in backgrounds. The inflorescence counter produced counts for each panel in the videos. In the first experiment, the average panel-wise counts and the total row-wise counts are given in Tables 4 and 5, respectively. A comparison of the ground-truth counts and the predicted counts is presented. To compare the counts, the normalised MAE (NMAE) error is calculated with NMAE $=$ MAE$/mean(y_i)$. For the average panel-wise counts, the minimum and maximum errors obtained are 3.46% and 5.88%, respectively. For the total row-wise counts, the minimum and maximum errors obtained are 2.10% and 5.89%, respectively. This shows an impressive counting performance with absolute errors less than 5.00% mostly. An estimate of the panel-wise counts within 5.00% error for different weather and background conditions shows the great potential of computer vision and deep learning-based approaches for yield estimation in viticulture.

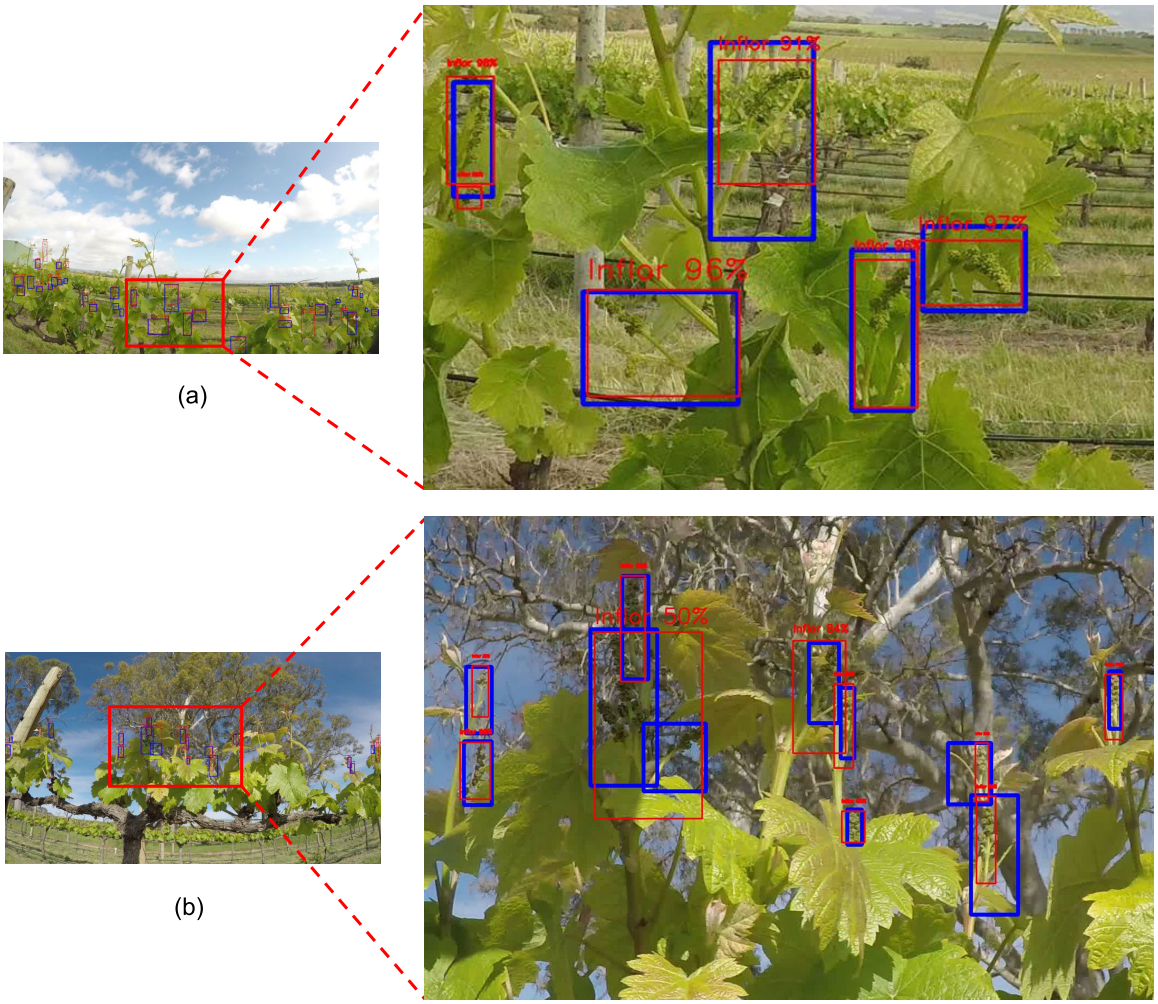


FIGURE 10. Inflorescence detection results on test video frames with different backgrounds. The ground-truth and detected bounding-boxes are represented in blue and red colours, respectively. The percentages above the bounding-boxes are the confidence scores. (a) A video frame with other rows in the background in good light. (b) A video frame with trees in the background in good light.

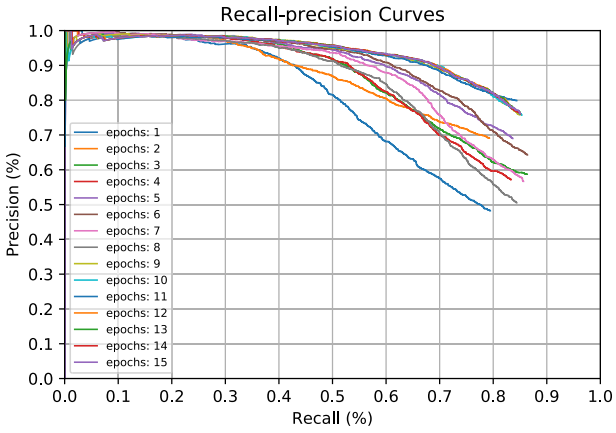


FIGURE 11. Recall-precision curves of the models trained up to 15 epochs.

In the second experiment, the panel-wise counts for the five test videos are analysed through scatter plots. For each test row/video, the inflorescence counter produced inflorescence counts for individual panels. The ground-truth and the predicted counts for each row using the panel-wise counts are shown in Fig. 13. The videos include different background variations, e.g., R83-east: the sun and blue-sky, R85-east: the sun and other rows, R86-west: trees, R87-west: trees and blue-sky, and R97-north: cloudy. The scatter plots show that for the test videos with different background variations, the predicted counts are quite close to the trend lines. In some cases, e.g., for sunny backgrounds (i.e., R83-east and R85-east), the trajectories tend to break and result in duplicate counting which then leads to over-estimating the counts. Moreover, for row R97-north, sometimes the counter has over-estimated the counts, and this is because *Shiraz* variety has a lot of inflorescences per panel and there are duplicate counts due to occlusions.

In the third experiment, the performance of the developed inflorescence tracker and counter has been analysed quantitatively through evaluation metrics, as described in

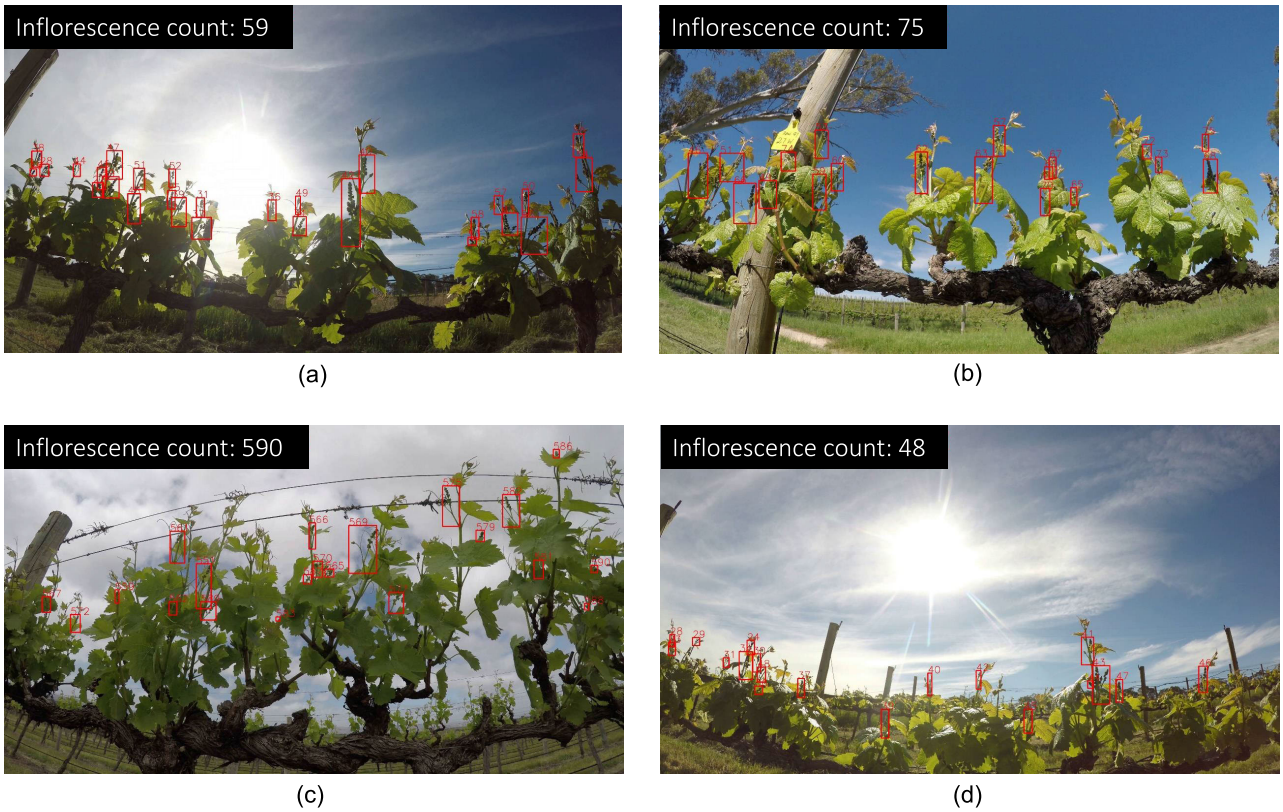


FIGURE 12. Inflorescence tracking and counting results in test videos, produced by our inflorescence tracker and counter for different variations in lighting and backgrounds. (a) the sun/exposure, (b) trees/blue-sky, (c) cloudy, and (d) the sun/other rows. The total counts produced up to a particular video frame are displayed in the top left corner of the frame.

Section IV-C. Different metrics including MAE, NMAE, RMSE, normalised RMSE (NRMSE) with NRMSE = RMSE/$mean(y_i)$, and R^2, were calculated using the panel-wise counts for the five test rows/videos. The metrics are shown in Table 6. For the row R83-east, MAE and RMSE of 8.44 and 10.08 inflorescences per panel were obtained, respectively. When these errors were normalised, NMAE and NRMSE of 9.32% and 11.13% per panel were obtained, respectively. The R^2 of 0.72 was obtained when comparing the ground-truth and predicted counts per panel. Similarly, these metrics were calculated for the other four rows. From these results, MAE ranges from 8.44 to 16.53 inflorescences per panel with percentage error ranging from 9.32% to 12.37%. Most of the time, we get R^2 more than 0.70. For all the panels combined in the five rows (Fig. 14), we achieve MAE and RMSE of 11.03 and 14.82 inflorescences per panel, respectively, NMAE and NRMSE of 10.80% and 14.50%, respectively, and R^2 of 0.86. Although these results show great potential, the viticulture industry is more interested in the average panel-wise counts and total row-wise counts, given in Tables 4 and 5, respectively, which give us a high-level and clear picture of the counting results rather than individual panel results.

In the fourth experiment, we discuss the panel densities and their relationship with the counting error. Panel density is a susceptible factor for tracking multiple inflorescences,

TABLE 6. Evaluation metrics calculated for each test row using the panel-wise ground-truth and predicted inflorescence counts.

Row	MAE	NMAE	RMSE	NRMSE	R^2
R83-east	8.44	9.32%	10.08	11.13%	0.72
R85-east	9.86	11.36%	14.05	17.04%	0.57
R86-west	7.92	9.90%	10.16	13.49%	0.71
R87-west	10.32	12.37%	12.12	15.43%	0.76
R97-north	16.53	10.91%	21.44	14.94%	0.70
Combined	11.03	10.80%	14.82	14.50%	0.86

as a dense panel would cause more occlusions and confusion in the association of inflorescences detected in consecutive video frames for tracking and counting. To observe the potential relationship of counting error with panel density, ground-truth versus predicted inflorescence counts per panel are analysed, as shown in Fig. 14. Here the auto counter refers to our developed inflorescence counter. The predicted counts by the auto counter are compared to the manual ground-truth, whereas the manual counter refers to manual counts produced by humans from one side of the row (predicted) and then compared with the other side of the row (ground-truth). The trend lines of the auto and manual counters almost coincide, which means that the auto counter can effectively approximate the manual counter in general. However, specifically, when the

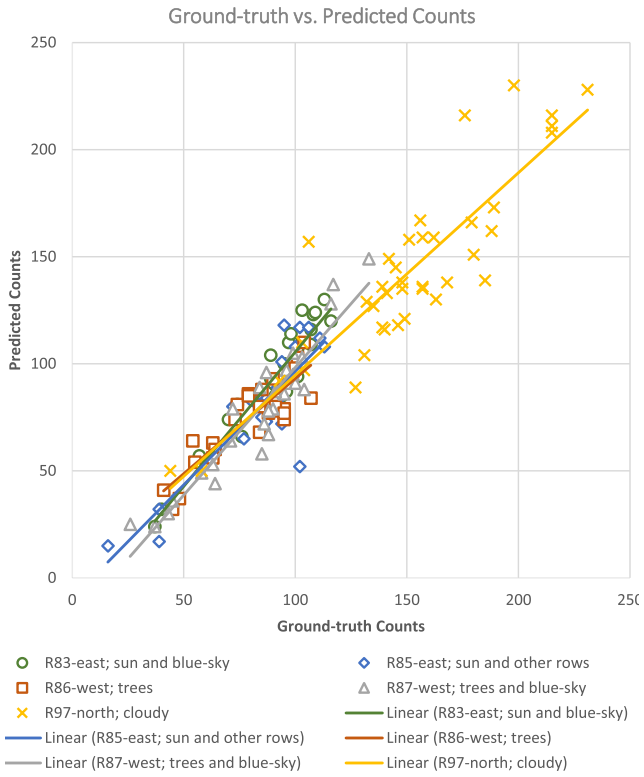


FIGURE 13. Ground-truth vs. predicted counts for the test rows/videos with different background variations.

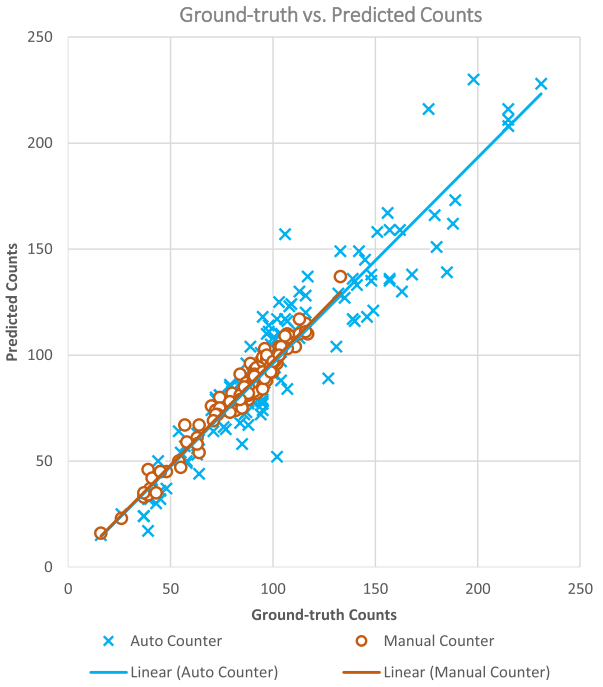


FIGURE 14. The relationships between counting error and panel density.

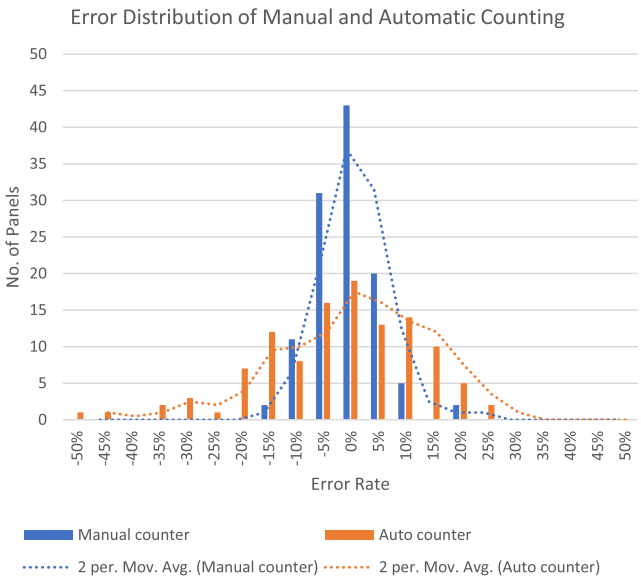


FIGURE 15. Error distribution of manual and automatic inflorescence counting.

panel density is low (less than 100 inflorescences per panel), the auto counter tends to under-estimate the counts. When the panel density is higher than 200 inflorescences per panel, the auto counter is more likely to over-estimate the inflorescence number. We have noticed that even manual counters cannot be perfect when counting inflorescences from two sides of the rows. Error distribution of the manual and auto counters are illustrated in Fig. 15 showing negative error for under-estimating and positive error for over-estimating the counts. Although the average error of the manual counter is lower than the auto counter (as expected), there is a certain percentage of error introduced during manual counting.

C. EARLY YIELD PREDICTION

Grapevine yield can be factored out into three components: the number of bunches (per vine, row or vineyard block), the number of berries per bunch, and the average berry weight. Berry weight is only able to be accurately assessed at harvest, as it can change throughout the maturation period, but is generally thought to have the lowest contribution to inter-seasonal variation. The berry number per bunch is determined by the number of individual flowers that set fruit. This occurs several weeks after bud-burst (October to November in the Southern hemisphere). The number of bunches is limited to the number of inflorescences that a vine produces and is typically considered to account for the largest proportion of inter-seasonal variation, around 60% [1]. Although some inflorescences can be produced on lateral cane growth during the growing season, these account for a very small proportion of harvested yield and are undesirable, and vineyard management generally attempts to prevent or remove these. The inflorescences on the main canes, those that grow at bud-burst, are present in the unopened buds as primordia and can be counted using dissection and a microscope. These buds are generated by the vine around anthesis (opening of the

flowers) in the preceding season, but most buds are pruned off the vine in Winter, so can only really be sampled for bud dissection shortly before the start of the growing season, following pruning. Not all buds will grow into new shoots and not all primordia will develop into inflorescences. Furthermore, there is no practical method foreseeable for on-the-go assessment of inflorescence primordia within unopened buds. Consequently, the earliest point in time that a yield can be predicted using data collected at scale is once the new inflorescences are visible, around growth stage E-L 12.

Neither berry weight nor berry number can be known at E-L 12 as the berries do not yet exist, thus multiplying the inflorescence number by the long-term average of bunch weight (berry number × berry weight) from previous seasons is the most straightforward method to generate a yield prediction. The potential accuracy of using this method was tested using data collected over three seasons on three different vineyard blocks to the ones used for video data acquisition at the McLaren Vale site. The method relies on a strong relationship between the inflorescence number at E-L 12 and the bunch number at harvest. Across the three blocks and seasons, this was indeed the case, with R^2 of this relationship varying between 0.66 and 0.96, and a mean R^2 of 0.79 (Fig. 16). The data also demonstrates the potential year-to-year variation in bunch number, with inflorescence and bunch counts in 2021/2022 being almost half those of 2019/2020 for example. The *Fiano* (another grapevine variety) inflorescence counts in 2019/2020 were higher than the bunch counts due to bunch thinning being carried out by the grower, in all other cases bunch counts were 10% to 15% higher than inflorescence counts due to bunches on laterals formed later in the season.

Yield predictions were made per panel as:

$$Y_{pred} = C_{infl} \times \overline{\mu}_{BWt_h} \times \overline{\mu}_{C_{bun}/C_{infl}}, \tag{7}$$

where Y_{pred} is the prediction, C_{infl} is the inflorescence count, $\overline{\mu}_{BWt_h}$ is the multi-season mean bunch weight for the block, and $\overline{\mu}_{C_{bun}/C_{infl}}$ is the multi-season relationship between the inflorescence count at E-L 12 and the bunch count at harvest. The resulting predictions were compared to the harvest yield on both a panel (Fig. 17) and a whole row (sum of all panels, Fig. 18) basis. For the nine rows (three blocks × three seasons) this method predicted yield with an R^2 of 0.97. For the panels within a single row and season ($n = 39$ in each case), R^2 values ranged from 0.07 to 0.83, with the lower values occurring when there was a little panel-to-panel variation. This situation did not affect the estimation at scale (row), with the per panel data that produced the lowest R^2 still providing a per-row prediction within 3% of the actual yield. The mean % error between the predicted and actual yield across the nine datasets was 8%, whereas if the yield was predicted using the three-year average yield for each block, the simplest form of yield prediction, the mean error was 28%.

Of the five rows/videos used for testing here, four were from the Woodside vineyard (i.e., rows R83-east, R85-east, R86-west, and R87-west). At that site, 60% of the inflorescences were removed as part of standard management practice in that block. Whilst crop thinning in this manner is not unusual, it is not practised over the majority of Australia's vineyard area and where it is practised, this extent of removal is very unusual. However, the result of this was that an estimate based on inflorescence counts and mean bunch weight would be 60% higher than the actual yield. When a yield estimate was generated from the predicted inflorescence counts produced by the auto counter and the average bunch weight across the block at harvest if historical data was not available, but block, rather than per row, bunch weights could be used to simulate some variation, this was indeed the case (Fig. 19). However, the row-to-row variation was maintained, despite crop thinning so a second yield estimate was generated for these rows by multiplying the predicted counts by 0.4, simulating the 60% crop thinning applied in practice. The McLaren Vale result (for the row R97-north) was generated using the predicted inflorescence count and a bunch weight estimated from the whole row harvest yield and a hand count of bunches prior to harvest, potentially less accurate than an actual harvest count. The average error of the early yield prediction across the four Woodside rows, using the adjusted estimate, was 4% below harvest yield and the error of the McLaren Vale row was 11% below harvest yield.

VI. CONCLUSION

Accurate counting of grapevine inflorescences in the field provides a mechanism to generate early yield estimations for the wine industry. In this work, four main contributions were made. First, a new video dataset was collected and curated for the inflorescence detection and counting tasks, as there were no public datasets available for these tasks. The inflorescences were annotated and manual counts were generated by domain experts to develop and evaluate the inflorescence detection and counting algorithms. A part of the dataset was made public for other researchers to work with. Second, an inflorescence detector was developed based on ResNeXt, FPN, and RetinaNet deep learning models. The detector can efficiently detect inflorescences in different lighting (e.g., bright and dark) and background (e.g., the sun, clouds, trees, and other rows) conditions. Visual and quantitative results for the inflorescence detection task are presented. For a five-fold cross-validation, the detector achieved an average precision of 80.00%, a recall of 83.92%, and an F1-score of 80.48%. Third, a KSP, FSM, and SiamesNet-based tracking algorithm was developed to automatically track the detected inflorescences and produce panel-wise and row-wise inflorescence counts. Five different videos of inflorescences of the *Shiraz* and *Tannat* varieties at two different vineyards were tested. The panel-wise counting results for all the videos were evaluated. An absolute error of 11.03 inflorescences per panel, an NMAE of 10.80%, and an R^2 value of 0.86 were obtained by the automatic inflorescence counter. Fourth, the counting results were used to generate an early yield estimate. The estimate was within 4% to 11% error in comparison with the actual yield after harvest and a broader analysis of hand counts and yield over multiple vineyards and seasons

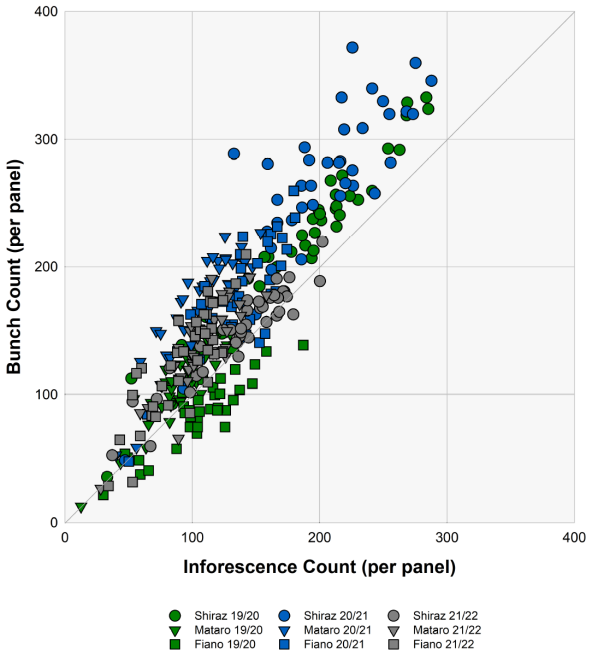


FIGURE 16. Relationship between inflorescence counts and bunch counts (per panel), for three different grapevine varieties across three consecutive seasons.

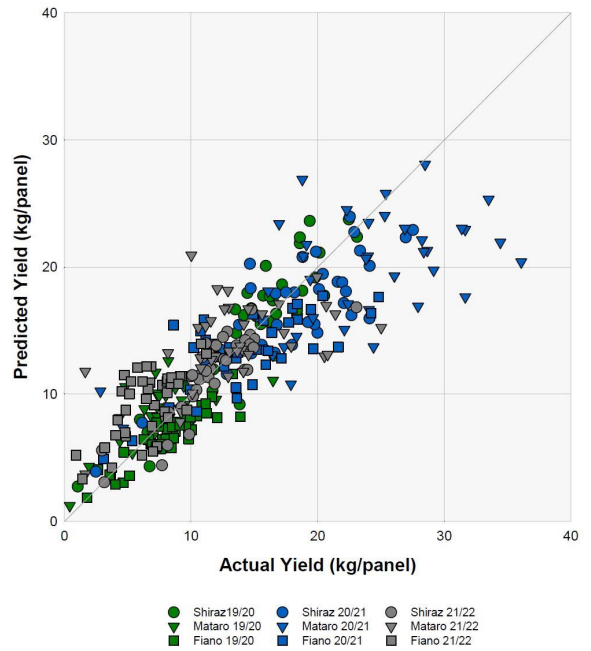


FIGURE 17. Actual yield versus predicted yield (kg per panel), for three different grapevine varieties across three consecutive seasons.

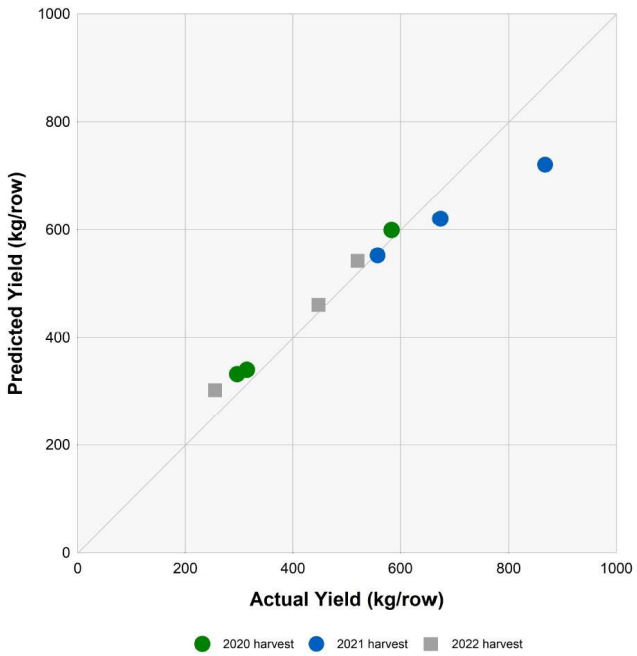


FIGURE 18. Actual yield versus predicted yield (kg per row), over three consecutive seasons.

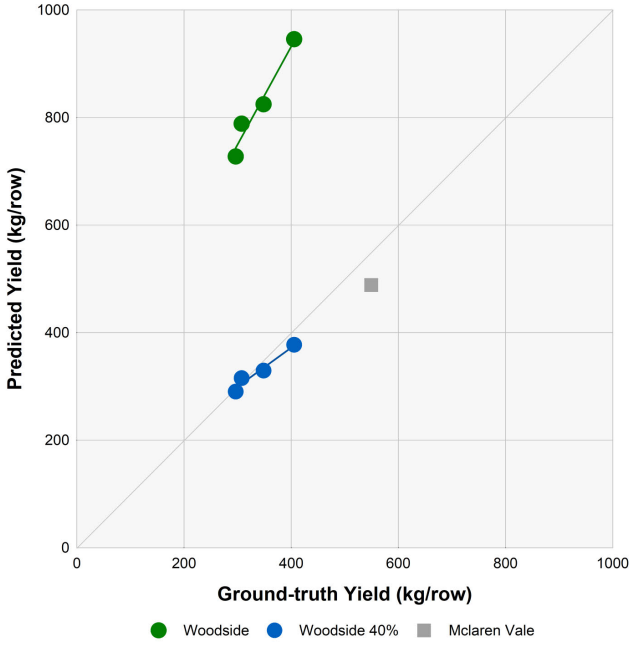


FIGURE 19. Yield estimates using proposed early yield prediction method.

suggests a potential average accuracy of around 10%, compared with almost 30% when simply using long-term production averages. In conclusion, the results demonstrate that deep learning and computer vision techniques can be used to develop improved decision-support tools for the viticulture industry.

APPENDIX A MOSAIC IMAGE FROM VIDEOS

In an attempt to count inflorescences in video data, an image mosaic based approach was also explored during this work. The idea was that the inflorescences detected within a mosaic image generated from a test video can represent the total number of inflorescences present in that video. For this task, a mosaic image of size 108,068 × 1,024 was created

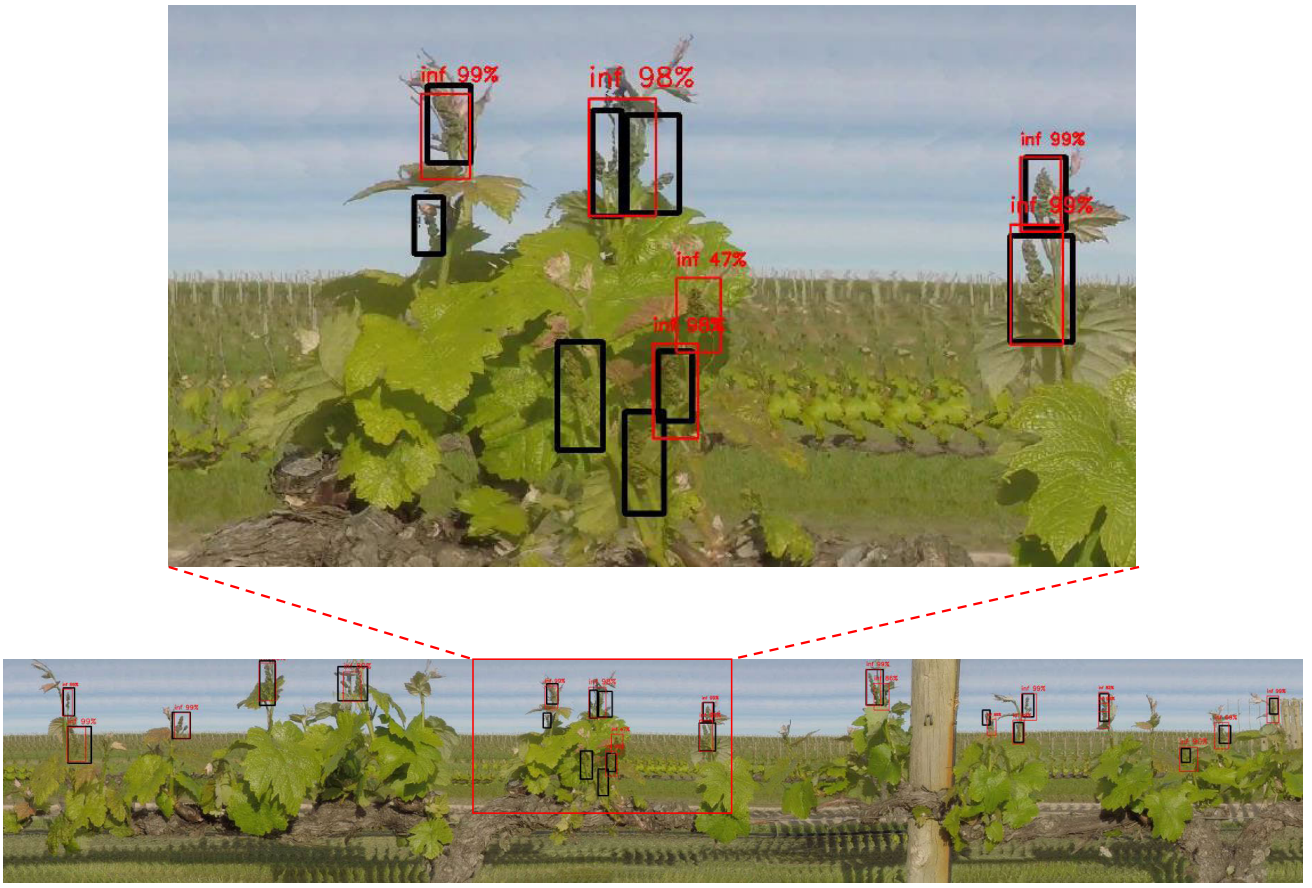


FIGURE 20. A segment of the mosaic image showing the inflorescence detection results. The ground-truth and the detected inflorescences are represented in black and red colours, respectively. The percentages above the bounding-boxes are the confidence scores.

from a test video using image stitching on the video frames. A sample segment of the mosaic image is shown in Fig. 20. Due to the image stitching, there are some image artefacts in the mosaic image such as overlapping regions. This becomes another challenge in detecting inflorescences, but it can save the effort of tracking the inflorescences to produce the counts.

The detection results on the mosaic image are demonstrated in Fig. 20. The ground-truth and the detected inflorescences are shown using black and red-coloured bounding-boxes, respectively, along with the confidence scores obtained for each detection. The inflorescences with clear backgrounds such as the sky were not affected much by the artefacts, whereas the inflorescences with complex backgrounds (leaves and other rows) were affected. In addition, some inflorescences disappeared in the mosaic image which were originally present in the video. This was because of the fixed viewing angle used during the image stitching. Furthermore, the FPs to TPs ratio was 38% in the mosaic image with no more options to reduce the FPs as compared to the detection and tracking approach. The algorithms for creating mosaic images from the videos can be improved further if this approach needs to be further explored in future.

ACKNOWLEDGMENT

The authors would like to acknowledge Adam Loveys (Woodside) and Accolade Wine (McLaren Vale) and their staff for their assistance and access to their vineyards. They also gratefully acknowledge the assistance of a number of CSIRO technical staff with fieldwork and image annotation. Finally, we acknowledge Wine Australia who invests in and manages research, development, and extension on behalf of Australia's grape growers and winemakers and the Australian Government.

REFERENCES

[1] P. R. Clingeleffer, K. J. Sommer, M. P. Krstic, G. Small, and M. A. Welsh, "Winegrape crop prediction and management," *Austral. New Zealand Wine Ind. J.*, vol. 12, no. 4, pp. 354–359, 1997.
[2] K. P. Seng, L.-M. Ang, L. M. Schmidtke, and S. Y. Rogiers, "Computer vision and machine learning for viticulture technology," *IEEE Access*, vol. 6, pp. 67494–67510, 2018.
[3] R. Perez-Zavala, M. Torres-Torriti, F. A. Cheein, and G. Troni, "A pattern recognition strategy for visual grape bunch detection in vineyards," *Comput. Electron. Agricult.*, vol. 151, pp. 136–149, Aug. 2018.
[4] S. F. Di Gennaro, P. Toscano, P. Cinat, A. Berton, and A. Matese, "A low-cost and unsupervised image recognition methodology for yield estimation in a vineyard," *Frontiers Plant Sci.*, vol. 10, p. 559, May 2019.
[5] R. G. V. Bramley, "Precision viticulture: Managing vineyard variability for improved quality outcomes," in *Managing Wine Quality*, A. G. Reynolds, Ed., 2nd ed. Amsterdam, The Netherlands: Elsevier, 2022, ch. 12, pp. 541–586.

[6] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Comput. Electron. Agricult.*, vol. 147, pp. 70–90, Aug. 2018.
[7] J. Tardaguila, M. P. Diago, B. Millan, J. Blasco, S. Cubero, and N. Aleixos, "Applications of computer vision techniques in viticulture to assess canopy features, cluster morphology and berry size," in *Proc. Int. Workshop Vineyard Mechanization Grape Wine Quality*, 2013, pp. 77–84.
[8] B. Millan, S. Velasco-Forero, A. Aquino, and J. Tardaguila, "On-the-go grapevine yield estimation using image analysis and Boolean model," *J. Sensors*, vol. 2018, pp. 1–14, Dec. 2018.
[9] M. P. Diago, A. Sanz-Garcia, B. Millan, J. Blasco, and J. Tardaguila, "Assessment of flower number per inflorescence in grapevine by image analysis under field conditions," *J. Sci. Food Agricult.*, vol. 94, no. 10, pp. 1981–1987, 2014.
[10] A. Aquino, B. Millan, D. Gaston, M. P. Diago, and J. Tardaguila, "vitisFlower: Development and testing of a novel Android-smartphone application for assessing the number of grapevine flowers per inflorescence using artificial vision techniques," *Sensors*, vol. 15, no. 9, pp. 21204–21218, Aug. 2015.
[11] B. Millan, A. Aquino, M. P. Diago, and J. Tardaguila, "Image analysis-based modelling for flower number estimation in grapevine," *J. Sci. Food Agricult.*, vol. 97, no. 3, pp. 784–792, Feb. 2017.
[12] S. Liu, X. Li, H. Wu, B. Xin, J. Tang, P. R. Petrie, and M. Whitty, "A robust automated flower estimation system for grape vines," *Biosyst. Eng.*, vol. 172, pp. 110–123, Aug. 2018.
[13] A. Aquino, B. Millan, S. Gutierrez, and J. Tardeguila, "Grapevine flower estimation by applying artificial vision techniques on images with uncontrolled scene and multi-model analysis," *Comput. Electron. Agricult.*, vol. 119, pp. 92–104, Nov. 2015.
[14] R. Rudolph, K. Herzog, R. Topfer, and V. Steinhage, "Efficient identification, localization and quantification of grapevine inflorescences in unprepared field images using fully convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2018, pp. 1–11.
[15] F. Palacios, G. Bueno, J. Salido, M. P. Diago, I. Hernandez, and J. Tardaguila, "Automated grapevine flower detection and quantification method based on computer vision and deep learning from on-the-go imaging using a mobile sensing platform under field conditions," *Comput. Electron. Agricult.*, vol. 178, Nov. 2020, Art. no. 105796.
[16] U. F. Rahim, T. Utsumi, and H. Mineno, "Deep learning-based accurate grapevine inflorescence and flower quantification in unstructured vineyard images acquired using a mobile sensing platform," *Comput. Electron. Agricult.*, vol. 198, Jul. 2022, Art. no. 107088.
[17] K. Pahalawatta, J. Fourie, A. Parker, P. Carey, and A. Werner, "Detection and classification of opened and closed flowers in grape inflorescences using Mask R-CNN," in *Proc. 35th Int. Conf. Image Vis. Comput. (IVCNZ)*, Nov. 2020, pp. 1–6.
[18] W. Du, Y. Zhu, S. Li, and P. Liu, "Spikelets detection of table grape before thinning based on improved YOLOV5s and K-means under the complex environment," *Comput. Electron. Agricult.*, vol. 203, Dec. 2022, Art. no. 107432.
[19] B. G. Coombe, "Growth stages of the grapevine: Adoption of a system for identifying grapevine growth stages," *Austral. J. Grape Wine Res.*, vol. 1, no. 2, pp. 104–110, 1995.
[20] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020, doi: 10.1109/TPAMI.2018.2858826.
[21] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural Comput.*, vol. 29, no. 9, pp. 2352–2449, Sep. 2017.
[22] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: A brief review," *Comput. Intell. Neurosci.*, vol. 2018, pp. 1–13, Feb. 2018.
[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Sep. 2014, pp. 1–14.
[24] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
[25] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 5987–5995.
[26] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021.
[27] Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie, Y. Wei, J. Ning, Y. Cao, Z. Zhang, L. Dong, F. Wei, and B. Guo, "Swin Transformer V2: Scaling up capacity and resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 12009–12019.
[28] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Dec. 2014, pp. 1–16.
[29] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.
[30] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[31] T. Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 936–944.
[32] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," May 2019, pp. 6105–6114.
[33] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Oct. 2018, pp. 6154–6162.
[34] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang, C. C. Loy, and D. Lin, "Hybrid task cascade for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jan. 2019, pp. 4969–4978.
[35] J. Berclaz, F. Fleuret, E. Türetken, and P. Fua, "Multiple object tracking using K-shortest paths optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1806–1819, Sep. 2011.
[36] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5000–5008.
[37] B. Li, W. Wu, Z. Zhu, and J. Yan, "High performance visual tracking with Siamese region proposal network," in *Proc. CVPR*, Jun. 2018, pp. 8971–8980.
[38] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. S. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 1328–1338.
[39] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[40] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Res. Logistics Quart.*, vol. 2, nos. 1–2, pp. 83–97, Mar. 1955.
[41] P. Bankhead, M. B. Loughrey, J. A. Fernandez, Y. Dombrowski, D. G. McArt, P. D. Dunne, S. McQuaid, R. T. Gray, L. J. Murray, H. G. Coleman, J. A. James, M. Salto-Tellez, and P. W. Hamilton, "QuPath: Open source software for digital pathology image analysis," *Sci. Rep.*, vol. 7, Dec. 2017, Art. no. 16878.
[42] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, and Z. Zhang, "MMDetection: Open MMLab detection toolbox and benchmark," Jun. 2019, *arXiv:1906.07155*.

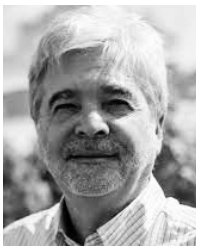
MUHAMMAD RIZWAN KHOKHER (Member, IEEE) received the B.S. degree in electrical engineering from International Islamic University Islamabad, Pakistan, in 2009, the M.S. degree in electrical engineering from the National University of Science and Technology, Pakistan, in 2012, and the Ph.D. degree from the School of Electrical, Computer, and Telecommunications Engineering, University of Wollongong, Australia, in 2018. In 2019, he joined CSIRO Data61, Australia, where he is currently a Research Scientist carrying out research and working on commercial applied projects. His research interests include image/video processing, computer vision, machine/deep learning, and artificial intelligence.

QIYU LIAO received the M.S. degree in signal and information processing from the University of Science and Technology of China and the Ph.D. degree in engineering and information technology from the University of Technology, Sydney. He is currently a Research Scientist with CSIRO Data61. His research interests include computer vision, deep learning, visual categorization, and multiple object tracking.

ADAM L. SMITH was born in Guyra, Australia, in 1990. He received the bachelor's degree (Hons.) in mechatronics engineering from The University of Newcastle, Australia, in 2017. From 2018 to 2022, he was a Research Technician with CSIRO, Adelaide, involved in digital viticulture projects, with the core work as sensor deployment and a focus on computer vision. His other research and development experience includes "Hone," a start-up located within the HMRI facility in Newcastle, building lab equipment, and Elite Robotics (another Newcastle start-up) where he designed computer vision modules for robot navigation.

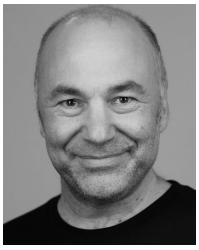
CHANGMING SUN received the Ph.D. degree in computer vision from Imperial College London, London, U.K., in 1992. He then joined CSIRO, Sydney, Australia, where he is currently a Principal Research Scientist carrying out research and working on applied projects. He is also a Conjoint Professor with the School of Computer Science and Engineering, University of New South Wales. He has served on the program/organizing committees for various international conferences. His current research interests include computer vision, image analysis, and pattern recognition. He is an Associate Editor of the *EURASIP Journal on Image and Video Processing*.

DONALD MACKENZIE graduated as a Science Technician from the South Australian Institute of Technology, in 1975. He was employed as a Technician with Flinders University and a Technical Adviser with Australian Government Foreign Aid (AUSAID) in Indonesian universities, in Java, Bali, Lombokm and West Timor, from 1972 to 1995. From 1998 to 2022, he involved in grapevine research as an Experimental Scientist with CSIRO Agriculture and Food, Waite Campus, Urrbrae, SA, USA.



MARK R. THOMAS was a Project Leader with CSIRO Agriculture and Food with more than 25 years of experience in grapevine genetics, breeding, and evaluation. From 2002 to 2008, he was the Chairperson of the International Grapevine Genome. He has published a number of leading articles and reviews on grapevine genetics and improvement, flowering, and grapevine germplasm resources. He received the PL Goldacre Medal from the Australian Society of Plant Physiologists, in 1994.

DADONG WANG (Senior Member, IEEE) received the D.Eng. degree from the University of Science and Technology Beijing, China, in 1997, and the Ph.D. degree from the University of Wollongong, Australia, in 2002. He is currently a Principal Research Scientist and the Leader of the Quantitative Imaging Research Team, Commonwealth Scientific and Industrial Research Organisation (CSIRO), Australia. Prior to joining CSIRO, in 2005, he was involved in industry as a Software Engineer and a Systems Engineer for about six years. He is also a Conjoint Associate Professor with the University of New South Wales (UNSW), and an adjunct Professor with the University of Technology, Sydney (UTS). His main research interests include image analysis, computer vision, artificial intelligence, and machine learning. His research team has a track record in developing intelligent end-to-end imaging solutions and via collaboration with industry partners bring them to the market. The team was a recipient of the Research Achievement Awards by CSIRO, the Engineering Excellence Award by Engineers Australia, Research and Development (R&D) Category of NSW, Queensland, and ACT iAwards.



EVERARD J. EDWARDS received the B.Sc. degree (Hons.) in plant sciences from The University of Sheffield, in 1992, and the Ph.D. degree in post-harvest physiology of potatoes from Nottingham Trent University, in 1997. He started his career in the U.K. He was a Postdoctoral Fellow with the University of York and The Australian National University. Both positions examined climate change effects on plants, the former investigating the impact of soil warming on root growth, and the latter the interaction between elevated CO2 and phosphorous availability on nitrogen fixation. Since 2006, he has been applying his background in whole plant physiology to perennial horticulture at CSIRO. Much of this has been in optimizing winegrape management, including: developing sensor technologies and analytics for viticulture; improving the understanding of rootstock conferred traits, such as their role in controlling vigour and water use efficiency; examining the role of vine balance in determining fruit composition (in collaboration with the Charles Sturt University and NSW DPI); the long-term impact of deficit irrigation; and the interaction between temperature and water status during heat-waves. He has also been involved in a number of other collaborative projects investigating aspects of winegrape management, including climate change aspects, with institutions, such as The University of Adelaide, SARDI, and DEDJTR Victoria.