EasyChair Preprint

№ 6539

Comparison of Grape Flower Counting Using Patch-Based Instance Segmentation and Density-Based Estimation with Convolutional Neural Networks

Umme Fawzia Rahim, Tomoyoshi Utsumi and Hiroshi Mineno

September 4, 2021

Comparison of grape flower counting using patch-based instance segmentation and density-based estimation with Convolutional Neural Networks

ABSTRACT

Information on flower number per grapevine inflorescence is critical for grapevine genetic improvement, early yield estimation and vineyard management. Previous approaches to automize this process by traditional image processing techniques such as color and morphology analysis, have failed in the improvement of a universal system that can be applied to multiple grapevine cultivars during different growth stages under various illumination conditions. Deep neural networks present numerous opportunities for image-based plant phenotyping. In this study, we evaluated three deep learning-based approaches for automatic counting of flower numbers on inflorescence images, built on instance segmentation using Mask R-CNN, object density-map estimation using U-Net and patch-based instance segmentation using Mask R-CNN, respectively. The results were analyzed on a publicly available grapevine inflorescence dataset of 204 images of four different cultivars during various growth stages, providing a high diversity for inflorescence morphology. The algorithm, based on patch-based instance segmentation using Mask R-CNN, produced counting results highly correlated to manual counts ($R^2 = 0.96$). Practically constant MAPE values among different cultivars (from 5.50% to 8.45%), implying a high robustness in this method. Achieving the fastest counting (0.33 sec. per image of size 512 × 512) with slightly lower counting accuracy ($R^2 = 0.91$), the method based on object density-map turned out to be suitable for real-time flower counting systems.

Keywords: Computer vision-based phenotyping, flower counting, instance segmentation, object density map, convolutional neural network (CNN), grape yield estimation

1. INTRODUCTION

Flower number per inflorescence is one of the major traits in the scope of grapevine breeding, breeding research and vineyard management[1–4] . A low number of flowers per inflorescence might limit the berry yield of certain varieties under certain condition[5]. On the other hand, a high number of flowers per inflorescence along with a high fruit-set rate might develop too compact clusters, which are more inclined to fungal vineyard diseases[6]. For optimized vineyard management, i.e. yield adjustments by bunch thinning, early yield predictions between fruit set and veraison (beginning of grape ripening), are essential to achieve well-balanced leaf-area-to-fruit-ratios, which are required for desired grape quality and yield amount[7,8]. Consequently, there is a growing significance on the accurate assessment of the flower number for the breeding of new cultivars with an optimum reproductive performance and for an early yield prediction to assist viticulturists taking management decisions.

The traditional method for assessment of flower number in an inflorescence relies on manual counting , however this process becomes challenging and unpractical when large grapevine sets need to be analyzed[9]. Recently, several approaches for automatic estimation of flower number per inflorescence by image analysis have been explored, including the works of Diago et al., Aquino et al., Benmehaia et al., Millan et al., Liu et al., Rudolph et al.[10–15]. However, most of these works which are based on traditional color analysis and morphological processing are sensitive to illumination, size and morphology of inflorescence and hence are not robust to inflorescences from diverse cultivars and to variations in images taken in different yard. Efficient identification and counting of grapevine flowers hereby is the most challenging phenotyping task regarding the small size and dense distribution of single flowers within inflorescence, as well as wide variability in grapevine inflorescence morphology between cultivars, locations and seasons (see Figure 1). As discussed by Liu et al.[14], variability in grapevine inflorescence morphology like stem length, branching pattern and flower density affect automatic flower counting performance for a method developed based on traditional color thresholding and image feature descriptors.

In recent years, advanced computer vision techniques using deep convolutional neural networks (DCNN) have been largely adopted in crop detection[16] and plant phenotyping[17] researches because of the networks' ability to be invariant to different types of variations in the image data in complex agricultural environment. Deep instance segmentation-based methods such as Mask R-CNN[18] can be used for accurate counting of objects, but both their segmentation and counting performance declines when the object size decreases and objects have heavy intern-occlusion. Patch-based approach, where instead of processing the whole image at once smaller patches from the image are extracted and processed in sliding window fashion can make instance segmentation network potentially more robust to smaller objects. In contrast, counting methods using object density maps[19] bypass the difficulties caused when objects have dense distribution by avoiding explicit detection[20]. In an object density map, the integral over any region gives the estimated number of objects within the corresponding region in the image.

The purpose of this study is to evaluate efficiency of grapevine flower counting per inflorescence by instance segmentation using state-of-the-arts Mask R-CNN[18] and by density-based estimation using U-Net[21], a widely used Fully-convolutional network (FCN) for image segmentation, as well as to assess the competence of patch-based processing for enhancing instance segmentation performance. To this aim, we have analyzed and compared the performance of the suggested models on publicly available grapevine inflorescence dataset by Liu et al.[14], including diverse cultivars, different inflorescence development stages and different seasons. Our experiments demonstrated satisfactory results under all conditions, providing useful information for grapevine breeding research and early yield prediction.

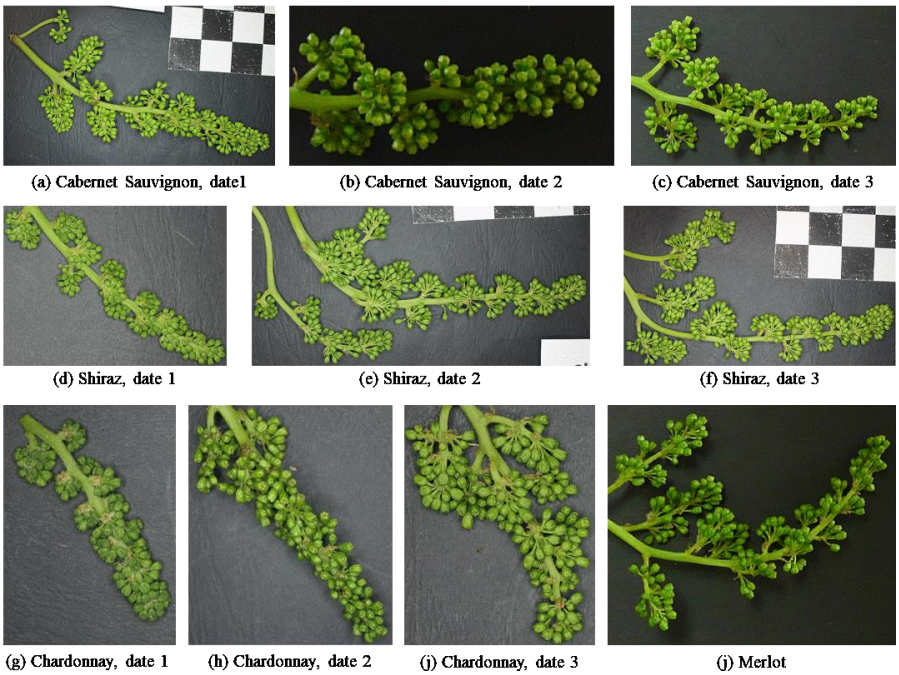

Figure 1. Challenges of automatic grapevine flower counting per inflorescence. Morphological diversity of inflorescences from different cultivars during different growth stages; the difference in density of flower distribution in an inflorescence, shape of the flowers and number of branches. Dates are referred in Table 1.

2. MATERIALS AND METHODS

2.1 Dataset preparation

For this study, we used grapevine inflorescence dataset by Liu et al.[14]. The dataset contains a total of 533 images of four grapevine cultivars (Cabernet sauvignon, Chardonnay, Merlot and Shiraz) captured under different illumination conditions at different Australian vineyards. These cultivars include a wide range of phenotypic diversity in flower inflorescence length, flower number, flower density and branch numbers, as observed in Figure 1. The growth stage of the sampled inflorescences ranged between Einhorn-Lorenz (EL) stage 12-21[22].

Since the images were captured by either commercial RGB camera or mobile there is no uniform image size or resolution in the dataset. We resized each image to 512 pixels in larger dimension while preserving the original aspect ratio and then pasted it on a black frame of 512 × 512 pixels, in order to make the images suitable for processing by both neural networks, Mask R-CNN and U-Net. The entire dataset was then sampled into three subsets for train, validation, and test purposes. The flower counting performances of the models were evaluated on a set of 204 images from all four grapevine cultivars and has the same images from the evaluation set of work [14]. From the remaining images 120 images were selected for training and another 25 images for validation. For performance evaluation and training the selected subsets were then annotated carefully using VGG image annotation tool. Each visible grape flower in the images were tagged by bounding polygons (see Figure 2a) and by points (see Figure 2b) for instance segmentation task and density map estimation task, respectively. The detail description of the datasets is presented in Table 1.

(a) (b)

Figure 2. Examples of manual annotation of individual flowers in an inflorescence image. Labeling flowers using bounding polygons for instance segmentation task (a) and using points for density map estimation task (b).

2.2 Automatic estimation of flower numbers

In this study, for automatic estimation of number of flowers per grapevine inflorescence three different approaches with two deep neural networks were evaluated and their efficiency were analyzed. Figure 3 illustrates our approaches for grape flower counting. Our first approach employed the state-of-the-arts instance segmentation network Mask-RCNN[18] for segmentation of each instance of flower in the inflorescence images and estimate their numbers simultaneously (Figure 3a). The segmentation-based method for counting requires explicit localization of each objects in the scene, therefore, it's performance potentially declines when the object size decreases and have dense distribution as such in the inflorescence datasets considered for experiments in this work. Counting using object density map avoids the explicit localization of objects and could produce promising results under conditions of smaller object size and heavy inter-occlusion. In our second approach, we used U-Net[21] semantic segmentation network to generate density map and estimate flower numbers on the inflorescence images (Figure 3b). In our final approach, we utilized patch-based processing technique with an aim to enhance the instance segmentation performance of Mask R-CNN in situations like our experimental dataset, where objects have smaller size and appear partially because of heavy inter-occlusions (Figure 3c). Detail description of patch-based processing is given in section 2.4.

2.3 CNN architectures

The Mask RCNN[18] network is the combination of a Faster R-CNN object detector[23] and a FCN for semantic segmentation, producing a complete, end-to-end, instance segmentation output. The Faster R-CNN is also composed of two architectures: a region proposal network (RPN) and an object detector, the Fast R-CNN[24]. RPN generates Region of Interests (RoIs) by finding anchors in the feature space, rectangular boxes that may contain objects of interest. The Fast R-CNN consists of a softmax object classifier and a per-class bounding box regressor.

U-Net[21] is a widely used FCN for image segmentation, frequently applied to microscopic biomedical data. It has autoencoder-like structure, where an input image is processed by repeated application of a block of convolutional layers, followed by a pooling layer (down-sampling). This way the network encodes and compresses the key features of the

input image. The second portion of U-Net is symmetric, but pooling layers are replaced with up-sampling convolutional layers, so that the output dimensions match the size of the input image. The information from higher resolution layers in the down-sampling portion is passed to corresponding layers in the up-sampling portion, which allows to reuse learned higher level features to precisely decode contracted layers.

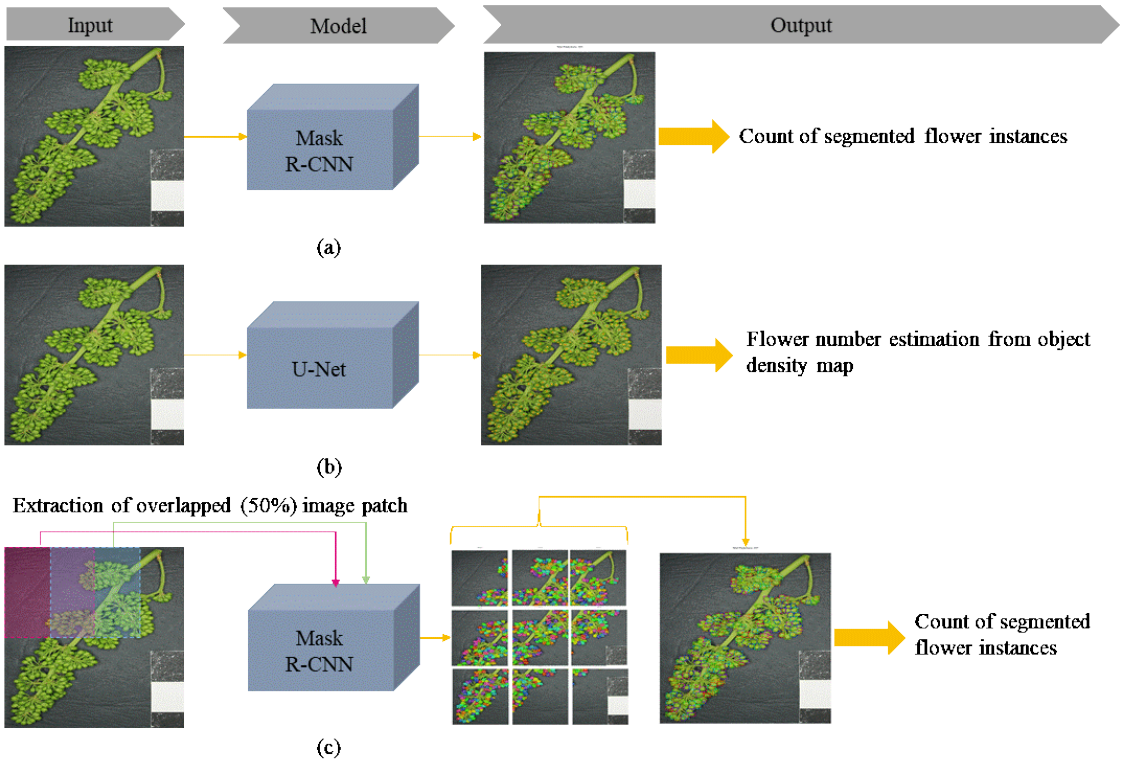

Figure 3. Illustration of the deep learning-based automatic approaches for estimation of flower numbers in grapevine inflorescence images.

2.4 Implementation, training and inference

For instance segmentation, we employed the publicly available Keras/TensorFlow-based implementation for Mask R-CNN by Matterport, Inc.[25], pre-trained with the COCO dataset[26]. No layer was frozen during training, so all weights could be updated by the training on the grapevine inflorescence datasets. ResNet-101 was used as feature extraction architecture. Our implementation of U-Net for density map estimation is also based on Keras/TensorFlow. We customized Matterport's implementation of Mask R-CNN for patch-based processing for our final image analysis approach.

The training set (see Table 1) was augmented 9 times: for each image in the training set, 8 augmentations were generated using horizontal flips, vertical flips, rotation (-90^0, 90^0, -180^0, 180^0), scaling (between 0.5 and 1.5 percent) and crop and pad (between -0.05 and 0.1 percent) using imgaug library[27]. The augmentations were applied randomly and repeatedly until a final set of 1080 images was obtained and used while training. Pixel level instance masks and point masks were used for training Mask R-CNN and U-Net, respectively.

For patch-based approach, each image in the training set was split into image patches of 256 × 256 pixels (mask images were produced accordingly) and used during training Mask R-CNN. During inference, 50% overlapped (horizontally and vertically) image patches of size 256 × 256 pixels were extracted from the full image and processed by the trained Mask R-CNN. A full confidence score image was produced by combining the confidence score patches and averaging the values of the overlapped regions between patches.

2.5 Evaluation and statistical analysis

The quality of the estimation derived by the models was assessed by comparing the number of visible flowers obtained automatically with the number obtained by manual counting (ground truth) on the images from the test set. The coefficient of determination (R^2), the Root Mean Square Error (RMSE) and mean absolute percentage error (MAPE) metrics according to Paulus et al.[28] were calculated for accuracy assessment and error estimation.

Table 1. Description of the datasets used in this paper. CAS, CHA, MER, and SHI are short for Cabernet Sauvignon, Chardonnay, Merlot, and Shiraz, respectively; EL is short for EL stage, the phenological grapevine growth stage according to the modified Einhorn-Lorenz system[22].

Purpose	Dataset	Date	Selected images	EL stage	Location	Total
Training	CAS, date 2	10/01/2015	15	16-17	Coonawarra	120
	CAS, date 3	12/01/2015		17		
	CAS, date 4	28/10/2014		15-16	Padthaway	
	CHA, date 4	23/10/2015		17-18	Barossa	
	CHA, date 5	12/01/2015		16	Coonawarra	
	CHA, date 6	23/10/2015		17-18	Barossa	
	SHI, date 4	12/01/2015		16	Coonawarra	
	SHI, date 5	22/10/2013		16-17	Barossa	
Validation	CAS, date 5	28/10/2013	10	16-17	Barossa	25
	CHA, date 7	15/10/2013		15		
	MER	12/01/2015	5	16-17	Coonawarra	
Evaluation	CAS, date 1	02/11/2015	25	17 -18	Barossa	204
	CHA, date 1	14/10/2015	22	14		
	CHA, date 2	20/10/2015	26	16-17		
	CHA, date 3	23/10/2015	29	19-20		
	MER	12/01/2015	20	16-17	Coonawarra	
	SHI, date 1	20/10/2015	27	15	Barossa	
	SHI, date 2	23/10/2015	30	17		
	SHI, date 3	28/10/2015	25	20		

To evaluate the performance of flower detection (segmentation) recall and precision were calculated based on the confusion matrix[29]. We also calculated the F1 score to assess the accuracy of visible flower detection (segmentation). Recall (r) and precision (p) measure the percentage of the actual flowers detected, and the percentage of flowers correctly identified, respectively, and is derived from

$$r = \frac{TP}{TP+FN}$$ and

$$p = \frac{TP}{TP+FP}$$

Where TP is the true positive value, indicating the number of flowers detected corresponding to an actual flower in the ground truth set, FN is the false negative, representing an actual flower, tagged in ground truth set but not detected by the automatic flower detection model, FP is the false positive, standing for the number of flowers detected that does not correspond to an actual flower in the ground truth set.

F1 score is derived by combining recall and precision,

$$F1 = \frac{2 \times r \times p}{r + p}$$

3. RESULTS AND DISCUSSION

The validation set was utilized to select the best model for further evaluation on the test set (see Table 1). All the networks were trained and tested on GPU (Nvidia GeForce GTX 1080 Ti) with a machine having Intel ® core i7-9700k 3.60 GHz processor and 64 GB RAM.

3.1 Comparison of different flower number estimation models

In this section, the performances of estimation of flower numbers per grapevine inflorescence obtained using instance segmentation with Mask R-CNN, using object density-map with U-Net and patch-based instance segmentation with Mask R-CNN were analyzed on our test set which includes inflorescence images from four different cultivars during multiple growth stages. For Mask R-CNN with patch-based processing, we obtained a significantly high R^2 value of 0.96 between the number of flowers obtained automatically and the number obtained by manual counting on the images, with individual values nearly matching the x=y identity line (Figure 4), and extremely small RMSE and MAPE of 12.77 flowers and 6.92%, respectively (Table 2). Considering each cultivar independently, the Mask R-CNN model with patch-based processing obtained R^2 values ranging between 0.99 (Cabernet sauvignon) and 0.94 (Chardonnay), values are slightly higher to those obtained by density-based estimation using U-Net (0.97, 0.90, 0.97, 0.91 for Cabernet sauvignon, Chardonnay, Merlot, and Shiraz, respectively) and much higher than to those achieved by Mask R-CNN model (0.70, 0.82, 0.57, 0.81 for Cabernet sauvignon, Chardonnay, Merlot, and Shiraz, respectively) (Table 2). Since R^2 is highly sensitive to the presence of outliers, we reported two more robust metrics for model performance evaluation: RMSE and MAPE (Table 2). The patch-based Mask R-CNN model showed lowest counting errors with RMSE ranged between 10.35 flowers (Shiraz) and 17.63 flowers (Cabernet sauvignon) and MAPE ranged between 5.50% (Shiraz) and 8.45% (Cabernet sauvignon) than density-based estimation (RMSE and MAPE from 15.78 to 24.31 flowers and from 6.78% to 15.08%, respectively) and ordinary Mask R-CNN (RMSE and MAPE from 42.75 to 69.90 flowers and from 20.98% to 27.38%, respectively).

The practically constant values of RMSE and MAPE among different cultivars produced by Mask R-CNN with patch-based processing and density-based estimation with U-Net is suggesting a high robustness in these automatic flower counting methods.

Figure 5a and Figure 5b show some examples of flower detection and counting results by ordinary Mask R-CNN (confidence threshold is 0.8), by density-map with U-Net and by patch-based Mask R-CNN (confidence threshold is 0.8) in three different columns. Overall, Mask R-CNN with patch-based processing detected almost all visible flowers correctly and therefore produced highly accurate counting results (Figure 5a column 2). We observed that the greatest counting errors found in the images with the highest number of flowers per inflorescence and heavy-inter occlusion (Figure 5b row 1,2), and very small size of flowers (Figure 5b row 3). Patch-based processing considerably enhanced the flower detection performance of Mask R-CNN (Figure 5a, b column 2 and column 1, respectively). However, despite of applying patch-based processing, the counting performance of Mask R-CNN was equivalent or slightly lower than density-based estimation in cases with heavy inter-occlusion and smaller flowers (Figure 5b), as Mask R-CNN failed to locate flowers under heavy occlusion and to detect small flowers.

The flower estimation model using density map with U-Net was much faster (0.33 sec. per image of size 512 × 512) than Mask R-CNN instance counting models (2.31 sec. and 3.83 sec. per image of size 512 × 512 for without and with patch-based processing, respectively. (Table 3), demonstrating a more suitability for real-time applications.

3.2 Comparison with baseline method

Table 4 presents the recall, precision and F1 score measured for four different cultivars. They demonstrate that the results produced by Mask R-CNN with patch-based processing has a substantially better performance than those reported in

previous related work by Liu et al.[14]. Given the automatic flower counting method using Mask R-CNN with patch-based processing has been tested on different cultivars at different development stages and showed consistent F1 scores (Table 4) and R^2 values (Table 5), respectively, the robustness of the developed method provides a solid foundation for flower number estimation and breeding research.

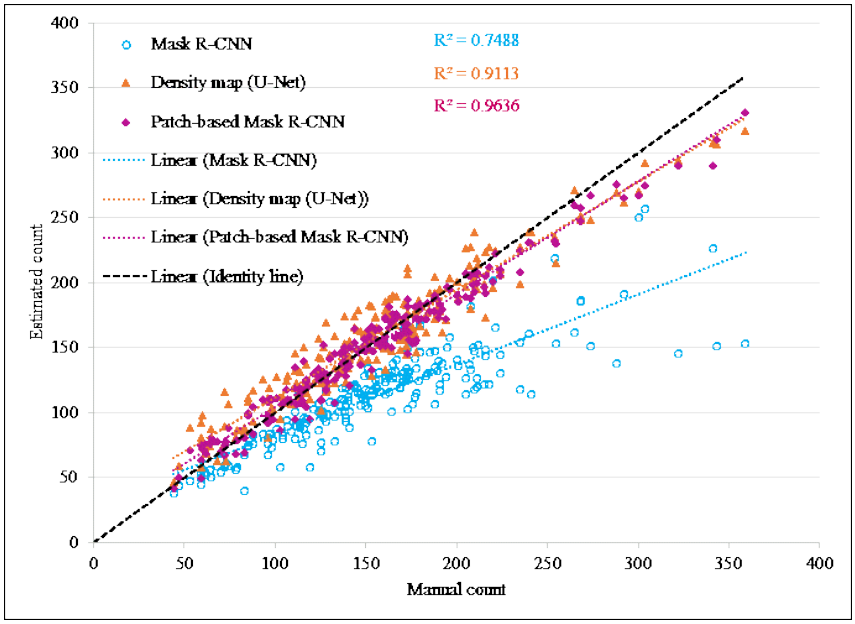

Figure 4. Correlation analysis to compare the number of flowers obtained manually and automatically by different estimation models.

Table 2. Performance of the different methods for visible flower counting in diverse image datasets of grapevine inflorescence. R^2, coefficient of determination; RMSE, root mean square error; MAPE, mean absolute percentage error.

Metrics	Methods	All cultivars	Cabernet sauvignon	Chardonnay	Merlot	Shiraz
R^2	Mask R-CNN	0.75	0.7029	0.8223	0.5678	0.8137
	Density map (U-Net)	0.91	0.9715	0.899	0.9754	0.9116
	Patch-based Mask R-CNN	**0.96**	0.99	0.9413	0.9762	0.9712
MAPE (%)	Mask R-CNN	25.17	23.51	24.38	20.98	27.38
	Density map (U-Net)	10.62	15.08	12.77	6.78	8.26
	Patch-based Mask R-CNN	**6.92**	8.45	8.12	6.53	5.50
RMSE (flowers)	Mask R-CNN	53.31	69.90	42.75	56.85	55.47
	Density map (U-Net)	18.11	24.31	18.39	16.81	15.78
	Patch-based Mask R-CNN	**12.77**	17.63	11.76	17.41	10.35

Table 3. Grapevine flower number estimation time per image of different methods.

Methods	No. of images	Image size (pixels)	Counting time/image (Sec.)
Mask R-CNN	204	512 × 512	2.31
Density map (U-Net)			**0.33**
Patch-based Mask R-CNN			3.83

4. CONCLUSIONS

Precise counting of flower number in grapevine inflorescence is essential for grapevine breeding research and early estimation of crop yield. Nevertheless, previous works have failed in the development of a generic method that could be applied to inflorescences with diverse morphology. In this work, we have evaluated three deep learning-based approaches for automatic counting of flower numbers employing instance segmentation, density-map estimation, and patch-based processing. The results were analyzed on a publicly available dataset that provide a high diversity for inflorescence morphology. Our results indicate that substantially high-throughput prediction of the visible number of flowers can be obtained utilizing advanced computer vision techniques, even when inflorescences with highly diverse morphology are investigated.

Table 4. Comparison of grapevine flower detection performance between the developed method based on patch-based Mask R-CNN and the results presented by Liu et al.[10] in terms of Precision, Recall, and F1 score.

Metrics	Method	Chardonnay			Shiraz			Cabernet sauvignon	Merlot
		Date 1	Date 2	Date 3	Date 1	Date 2	Date 3		
Precision	Patch-based Mask R-CNN	**0.9927**	**0.9779**	**0.9852**	**0.9886**	**0.9857**	**0.9776**	**0.9733**	**0.9867**
	Liu at el.[14]	0.9512	0.9352	0.9241	0.9228	0.8904	0.8788	0.8996	0.8916
Recall	Patch-based Mask R-CNN	0.9248	**0.9698**	**0.9747**	**0.9374**	**0.9694**	**0.9820**	**0.9590**	0.9583
	Liu at el.[14]	**0.9612**	0.9560	0.9611	0.9367	0.9262	0.9638	0.9173	**0.9819**
F1	Patch-based Mask R-CNN	**0.9575**	**0.9738**	**0.9799**	**0.9615**	**0.9775**	**0.9798**	**0.9661**	**0.9723**
	Liu at el.[14]	0.9555	0.9446	0.9417	0.9284	0.9069	0.9188	0.9069	0.9344

Table 5. Comparison of impact of inflorescence development stage on flower number estimation performance. R^2, coefficient of determination.

Metrics	Methods	Chardonnay, date 1	Chardonnay, date 2	Chardonnay, date 3	Shiraz, date 1	Shiraz, date 2	Shiraz, date 3
R^2	Patch-based Mask R-CNN	0.97	0.98	0.98	0.97	0.97	0.99
	Liu at el.[14]	0.79	0.93	0.94	0.82	0.94	0.90

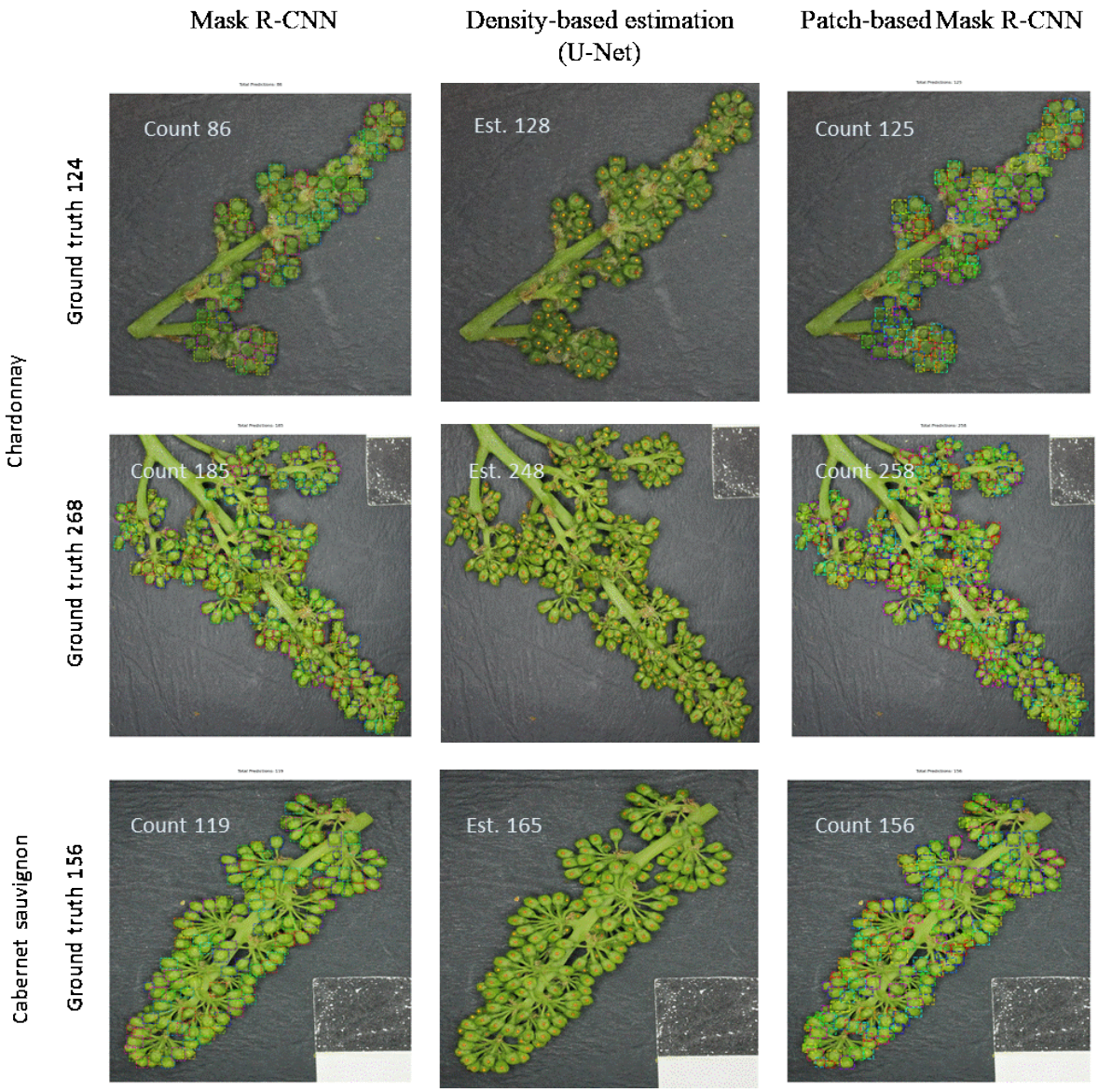

Figure 5a. Examples of flower counting results produced by Mask R-CNN (column 1), density-based estimation (column 2) and patch-based mask R-CNN (column 3).

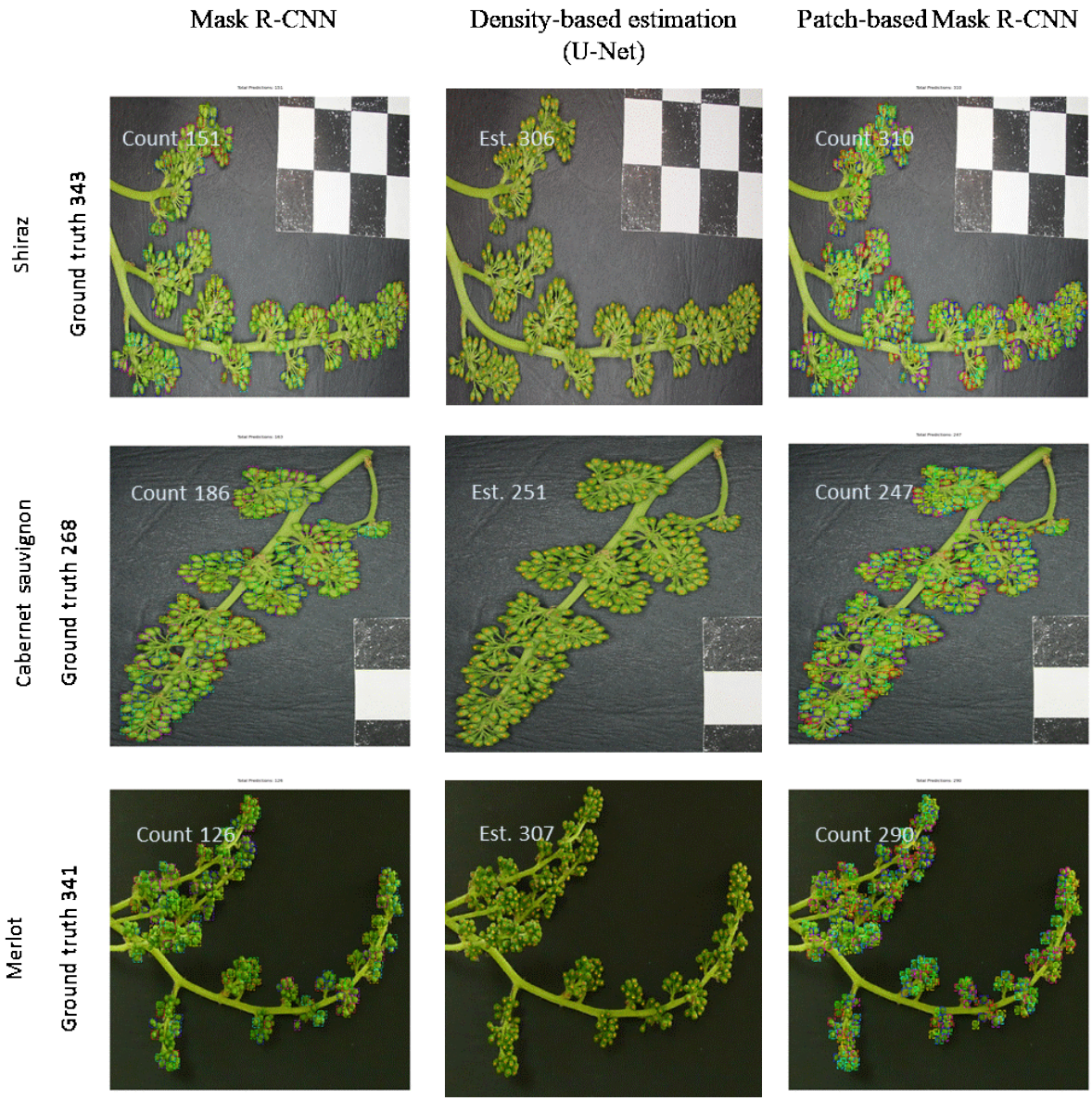

Figure 5b. Examples of flower counting results on images with the highest number of flowers per inflorescence produced by Mask R-CNN (column 1), density-based estimation (column 2) and patch-based mask R-CNN (column 3).

REFERENCES

[1] Molitor, D., Behr, M., Hoffmann, L. and Evers, D., "Impact of grape cluster division on cluster morphology and bunch rot epidemic," Am. J. Enol. Vitic. **63**(4), 508–514 (2012).

[2] Preszler, T., Schmit, T. M. and Heuvel, J. E. Vanden., "Cluster thinning reduces the economic sustainability of Riesling production," Am. J. Enol. Vitic. **64**(3), 333–341 (2013).

[3] Topfer, R. and Eibach, R., "Pests and diseases: Breeding the next-generation disease-resistant grapevine varieties," Wine Vitic. J. **31**(5), 47 (2016).

[4] Simonneau, T., Lebon, E., Coupel-Ledru, A., Marguerit, E., Rossdeutsch, L. and Ollat, N., "Adapting plant material to face water stress in vineyards: which physiological targets for an optimal control of plant water status?," OENO One **51**(2), 167 (2017).

[5] Dry, P. R., Longbottom, M. L., McLoughlin, S., Johnson, T. E. and Collins, C., "Classification of reproductive performance of ten winegrape varieties," Aust. J. Grape Wine Res. **16**, 47–55 (2010).

[6] Tello, J. and Ibáñez, J., "What do we know about grapevine bunch compactness? A state-of-the-art review," Aust. J. grape wine Res. **24**(1), 6–23 (2018).

[7] Auzmendi, I. and Holzapfel, B. P., "Leaf area to fruit weight ratios for maximising grape berry weight, sugar concentration and anthocyanin content during ripening," XXIX Int. Hortic. Congr. Hortic. Sustain. Lives, Livelihoods Landscapes IV 1115, 127–132 (2014).

[8] de la Fuente, M., Linares, R., Baeza, P., Miranda, C. and Lissarrague, J. R., "Comparison of different methods of grapevine yield prediction in the time window between fruitset and veraison," OENO One **49**(1), 27–35 (2015).

[9] Poni, S., Casalini, L., Bernizzoni, F., Civardi, S. and Intrieri, C., "Effects of early defoliation on shoot photosynthesis, yield components, and grape composition," Am. J. Enol. Vitic. **57**(4), 397–407 (2006).

[10] Diago, M. P., Sanz-Garcia, A., Millan, B., Blasco, J. and Tardaguila, J., "Assessment of flower number per inflorescence in grapevine by image analysis under field conditions," J. Sci. Food Agric. **94**(10), 1981–1987 (2014).

[11] Aquino, A., Millan, B., Gutiérrez, S. and Tardáguila, J., "Grapevine flower estimation by applying artificial vision techniques on images with uncontrolled scene and multi-model analysis," Comput. Electron. Agric. **119**, 92–104 (2015).

[12] Benmehaia, R., Khedidja, D. and Bentchikou, M. E. M., "Estimation of the flower buttons per inflorescences of grapevine (Vitis vinifera L.) by image auto-assessment processing," African J. Agric. Res. **11**(34), 3203–3209 (2016).

[13] Millan, B., Aquino, A., Diago, M. P. and Tardaguila, J., "Image analysis-based modelling for flower number estimation in grapevine," J. Sci. Food Agric. **97**(3), 784–792 (2017).

[14] Liu, S., Li, X., Wu, H., Xin, B., Petrie, P. R. and Whitty, M., "A robust automated flower estimation system for grape vines," Biosyst. Eng. **172** (2018).

[15] Rudolph, R., Herzog, K., Töpfer, R. and Steinhage, V., "Efficient identification, localization and quantification of grapevine inflorescences and flowers in unprepared field images using Fully Convolutional Networks," Vitis **58**(3), 95–104 (2019).

[16] Santos, T. T., de Souza, L. L., dos Santos, A. A. and Avila, S., "Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association," Comput. Electron. Agric. **170**, 1–22 (2020).

[17] Palacios, F., Bueno, G., Salido, J., Diago, M. P., Hernández, I. and Tardaguila, J., "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. Agric. **178**, 105796 (2020).

[18] He, K., Gkioxari, G., Dollár, P. and Girshick, R., "Mask r-cnn," Proc. IEEE Int. Conf. Comput. Vis., 2961–2969 (2017).

[19] Zhang, Y., Zhou, D., Chen, S., Gao, S. and Ma, Y., "Single-image crowd counting via multi-column convolutional neural network," Proc. IEEE Conf. Comput. Vis. pattern Recognit., 589–597 (2016).

[20] Kang, D., Ma, Z. and Chan, A. B., "Beyond counting: Comparisons of density maps for crowd analysis tasks—counting, detection, and tracking," IEEE Trans. Circuits Syst. Video Technol. **29**(5), 1408–1422 (2018).

[21] Ronneberger, O., Fischer, P. and Brox, T., "U-net: Convolutional networks for biomedical image segmentation," Int. Conf. Med. image Comput. Comput. Interv., 234–241, Springer (2015).

[22] Coombe, B. G., "Growth stages of the grapevine: adoption of a system for identifying grapevine growth stages," Aust. J. grape wine Res. **1**(2), 104–110 (1995).

[23] Ren, S., He, K., Girshick, R. and Sun, J., “Faster r-cnn: Towards real-time object detection with region proposal networks,” Adv. Neural Inf. Process. Syst., 91–99 (2015).

[24] Girshick, R., “Fast r-cnn,” Proc. IEEE Int. Conf. Comput. Vis., 1440–1448 (2015).

[25] Abdulla, W., “Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow; 2017,” Avaiblable at https://github. com/matterport/Mask_RCNN (2017).

[26] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C. L., “Microsoft coco: Common objects in context,” Eur. Conf. Comput. Vis., 740–755, Springer (2014).

[27] Jung, A. B., Wada, K., Crall, J., Tanaka, S., Graving, J., Yadav, S., Banerjee, J., Vecsei, G., Kraft, A. and Borovec, J., “Imgaug,” GitHub San Fr. CA, USA (2020).

[28] Paulus, S., Behmann, J., Mahlein, A.-K., Plümer, L. and Kuhlmann, H., “Low-cost 3D systems: suitable tools for plant phenotyping,” Sensors **14**(2), 3001–3018 (2014).

[29] Fawcett, T., “An introduction to ROC analysis,” Pattern Recognit. Lett. **27**(8), 861–874 (2006).