

Yield Estimation in Vineyards by Visual Grape Detection

Stephen Nuske, Supreeth Achar, Terry Bates, Srinivasa Narasimhan and Sanjiv Singh.

Abstract—The harvest yield in vineyards can vary significantly from year to year and also spatially within plots due to variations in climate, soil conditions and pests. Fine grained knowledge of crop yields would allow viticulturists to better manage their vineyards. The current industry practice for yield prediction is destructive, expensive and spatially sparse – small samples are taken from the vineyards during the growing season and extrapolated to determine overall yield. We present an automated method that uses computer vision to identify and count grape berries. These counts are used to generate per vine estimates of crop yield. Both shape and visual texture are used to detect berries. We demonstrate detection of green berries against a green leaf background. We present crop yield estimation results, with the actual harvest yield as groundtruth for 200 vines (over 450 meters) of two different grape varieties. We calibrate our berry count to yield and find that we can predict yield to within 9.8% of actual crop weight.

I. INTRODUCTION

Predicting the eventual weight of the harvest yield in a vineyard enables vineyard managers to make adjustments to the vines to reach their yield and quality goals. The current industry practice for predicting harvest yield is labor intensive, expensive, inaccurate, spatially sparse, destructive and riddled with subjective inputs. Typically, the process for yield prediction is for workers to sample a certain percentage of the vineyard and extrapolate these measurements to the entire vineyard. The weight of the clusters is constantly increasing until harvest, so the vineyard manager must guess at what percentage of the final weight is the current measurement – a subjective input which leads to inaccurate predictions. The manual sampling practice scales poorly to large commercial vineyards and the industry is searching for an alternative.

Here we report results of an approach to automatically detect and count grapes to forecast yield with precision and accuracy. Our approach is to take conventional visible light cameras through a vineyard to image the vines and detect the crop and predict yield. Traditional manual yield estimates look to sample the average number of grape clusters per-vine, the average number of grape berries per-cluster and average berry weight. Our approach is to estimate the total number of berries, essentially combining clusters per-vine and berries per cluster in the one measurement. Clusters per vine and berries per cluster account for 60% and 30% of variation in yield per vine respectively, therefore 90% of the variation in yield is accounted with accurate berry counts. Furthermore, the number of berries per-vine is a good measure to obtain

Stephen Nuske, Supreeth Achar, Srinivasa Narasimhan and Sanjiv Singh at Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA, 15213 nuske@cmu.edu

Terry Bates at Cornell University, 6592 West Main Road, Portland, NY, USA, 14769 trb7@cornell.edu



Fig. 1. Example camera image of Gerwurztraminer wine grapes captured at véraison. Automatically detecting the grape crop within imagery such as this is difficult because of issues caused by the lighting and shadows, and the lack of contrast to the leaf background.

because it is fixed from fruit-set all the way until harvest, unlike cluster weight for which a multiplier must be guessed and applied.

The challenges in visually detecting grape berries is their varying appearance under different lighting, the lack of color contrast to the background, which is often similarly colored to the grapes, and also occlusions causing not all grapes to be visible. An example of the difficulties of visually detecting grape crop can be seen in Fig. 1. The few existing methods for detecting crop in vineyards, have been restricted to the laboratory [4] or have relied on color contrast [3] and are therefore not applicable for detecting crop over a similarly colored background of leaves. Lack of color contrast is an important issue that occurs in the white-grape varieties and all the grape varieties prior to véraison (the onset of color development). We specifically address the issues of lighting and lack of color contrast, by using shape and texture cues for detection.

The issue of occlusion means it is not possible to detect and count all berries on a vine. However, our detection of grape berries is precise, ensuring that there are very few false positives. The result of precise detection is that our berry count is a reliable measurement of yield, despite the fact that our algorithm only counts a percentage of all the grape berries on a vine. We calibrate our berry count measurement to harvest yield from a set of vines, and apply this calibration to other vines not included in the calibration set, pointing to the fact that percentage of berries not detected is relatively constant from vine to vine.

We deployed our method in a vineyard and conducted an experiment in which manual per-vine harvest weights were collected and used as ground truth to evaluate our

automated yield measurements. The size of the experiment is significant, including roughly 450m of vines, including two different grape varieties, where the total harvest weight of the vines totaled over 2000kg. Our method calculates yield within 9.8% of ground truth.

II. RELATED WORK

Current practices to forecast yield are inaccurate because of sampling approaches that tend to adjust towards historical yields and include subjective inputs (Clingeffer et al. [2]). The calculation of final cluster weight from weights at véraison use fixed multipliers from historic measurements, Wolpert and Vilas [11]. Unfortunately, multipliers are biased towards healthier vines thus discriminating against missing or weak vines and multipliers for cluster weights vary widely by vineyard, season and variety.

Sensor-based yield estimation in vineyards has been attempted with trellis tension monitors, multispectral sensors, terahertz-wave imaging and visible-light image processing. A dynamic yield estimation system based on trellis tension monitors has been demonstrated (Blom and Tarara [1]) but it requires permanent infrastructure to be installed. Information obtained from multispectral images has been used to forecast yields with good results but is limited to vineyards with uniformity requirements (Martinez-Casasnovas and Bordes [8]). A proof of concept study by Federici et al. [4] has shown that terahertz imaging can detect the curved surfaces of grapes and also has the potential to detect these through occluding thin canopy. The challenge for this approach is to achieve fast scan rates to be able to deploy the scanner on a mobile platform.

Small scale yield estimation based on simple image color discrimination has been developed by Dunn and Martin [3]. This approach was attempted on Shiraz post-véraison (i.e. after color development, very close to harvest) in short row segments. The method would not be applicable for the majority of real world examples where the fruit appears over a background of similarly-colored leaves, as is the case in white grape varieties and in all varieties before véraison. More complex crop detection based on computer vision methods using color pixel classification or shape analysis has been attempted on various fruit types – Jimenez et al. [5] provides a summary of fruit detection work, Singh et al. [9] present a method for detecting and classifying fruit in apple orchards and Swanson et al. [10] use the shading on the curved surfaces of oranges as a cue for detection.

III. BERRY DETECTION

We deploy a sideways-facing camera and lighting on a small vineyard utility vehicle. The images capture the vines and are processed with our algorithm to detect and count the crop. In traditional vineyard yield estimation the crop components that are measured to derive a final estimate are:

- 1) Number of clusters per vine (60% of the yield variation)
- 2) Number of berries per cluster (30% of the yield variation)

- 3) Berry size (10% of the yield variation)

These three components combine to describe all the variation in harvest yield. Current practice is to take samples of each of these components to compute an average and compute the final yield. We take an approach to estimate the first two of these items together in one measurement – that of the number of berries per vine. The reason being that it is difficult, especially late in the season, to delineate the boundaries of clusters within images. However, it is possible to count the total number of berries seen, hence combining the two components – number of clusters per vine and berries per cluster – into one measurement: berries per vine. An interesting observation can be drawn that humans are better at counting clusters per vine and weighing individual clusters, whereas conversely it seems robotic sensing struggles to accurately count mature grape clusters. Instead it is easier to use robotic sensing to count the number of berries on vine, a measure which would not be possible for a human to directly produce.

Our approach does not attempt to measure berry weight. However, we account for 90% of the harvest yield variation with berries per vine ([2]). Furthermore, instead of taking a small sample and extrapolating, we aim to estimate non-destructively the specific yield at high resolution across the entire vineyard. Hence, we will not introduce sampling errors into the process.

Our algorithm to detect the berries in imagery has three distinct stages:

- 1) Detecting potential berry locations with a radial symmetry transform (Section III-A)
- 2) Identifying the potential locations that have similar appearance to grape berries (Section III-B)
- 3) Group neighboring berries into clusters (Section III-C)

A. Detecting Potential Berry Locations with a Radial Symmetry Transform

The first step of our algorithm is to find points with a high level of radial symmetry as these points are potential centers for grape berries, see Fig. 2(a). To find these points, we use the radial symmetry transform of Loy and Zelinsky [7]. The algorithm is robust to the issues of lighting and low color contrast, which cause problems for the existing crop detection techniques that rely on simple color discrimination (Jimenez et al. [5], Dunn and Martin [3]). The approach detects the centers of berries of all colors, even those that are similarly colored to the background leaves.

The radial symmetry transform requires us to know the radii of the berries as seen in the image ahead of time. The berry radii (in pixels) are dependent on the focal length of the camera, actual berry size and the distance from the camera. The focal length is kept fixed in our tests and the vehicle maintains a relatively constant distance from the vines. There is still variation in the radius the berries appear in the image from differing berry sizes and also some variation in location within the vine. We account for this variation by searching for radially symmetric points over a range of possible radii, N . Individual radii are denoted as n .

The transform first computes the locally normalized gradient \mathbf{g} with magnitude and orientation information at each image pixel. An example gradient image from the Sobel transform is depicted in Fig. 3(a). In a Hough Transform like setup, each edge pixel \mathbf{p} , with a gradient value above a threshold T votes for possible points of radial symmetry $\mathbf{p}_s(\mathbf{p})$ given by:

$$\mathbf{p}_s(\mathbf{p}) = \mathbf{p} \pm n \frac{\mathbf{g}(\mathbf{p})}{\|\mathbf{g}(\mathbf{p})\|} \quad (1)$$

for each radius n , these votes from the edge pixels are counted in a vote image F_n which is then smoothed out with A_n , a 2D Gaussian filter, to produce S_n , the radial filter response at radius n . These filter responses at different radii are then combined to form the overall radial filter response S which is given by.

$$S_n = F_n * A_n \quad (2)$$

$$S = \max_{n \in N} S_n \quad (3)$$

We compute local maxima in the response image S with a non-maximal suppression, and threshold to find the potential centers. We choose the threshold to ensure as many berry centers are detected as possible, at the expense of many false positive detections. We use the following stages in the algorithm to filter out the false positives.

B. Classifying Interest Points Appearing Similar to Berries

The next stage in our algorithm is to classify the detected points which appear most like grapes, see Fig. 2(b). We first take a patch in the image around each detected center. The patch size has a radius defined by the previous radial symmetry detector step. We then compute features from that image patch. The features we use are a combination of color and texture filters, which combine to form a 34 dimensional feature vector. We use the three RGB channels, the three L*a*b color channels and Gabor filters with 4 scales and 6 orientations. The features are not chosen specifically for the grape detection task – we use generic low-level image features.

We take a small number of training samples from our datasets, by selecting a random subset of images and manually define in the images which regions have grape berries. We compute our features in these regions which correspond to the positive examples of the appearance of berries. For negative examples we compute features at radially symmetric interest points outside of our defined crop areas.

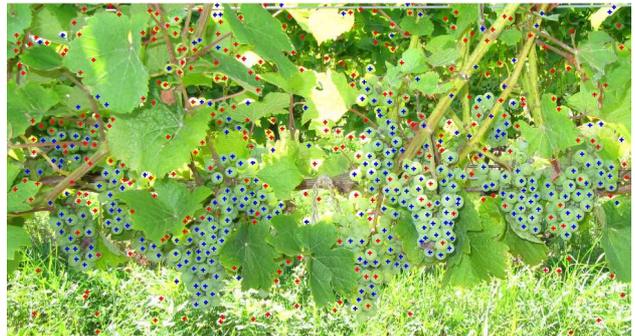
Given an input image we take each radially symmetric interest point, compute the feature vector, and apply the k-Nearest Neighbors algorithm. The k-Nearest Neighbors algorithm computes the distance in feature space to every point in the training set and determines whether the nearest neighbors are positive berry examples or negative. If the k closest positive examples are closer than the k closest negative examples, that interest point is classified as a berry. We use a value of three for k , which empirically seems to function appropriately.

C. Group Neighboring Berries into Clusters

After classification of the interest points, a small number of false positives still remain. Most of the remaining false positive detections are isolated while grape berries naturally occur in clusters so we apply contextual constraints that dictate that there should be a minimum number of berries in a cluster. We cycle through each classified berry, computing the distance to every other berry, and remove berries that do not have at least 5 other berries within their immediate neighborhood, which we define as a radius of 150 pixels. The process results in the clustered berries, which are the output of our entire algorithm, see Fig. 2(c).



(a) Detect Berry Locations with Radial Symmetry Transform



(b) Identify Locations with Similar Appearance to Grape Berries



(c) Group Neighborhoods of Berries into Clusters

Fig. 2. Example images showing the functioning of our visual berry detection algorithm on a Gerwurztraminer vine. Input image is seen in Fig 1. (a) potential berry locations in the image that have been detected as having radial symmetry. (b) points marked blue have been classified as having appearance similar to a berry. (c) classified berries that neighbor other classified berries are clustered together.

TABLE I

BERRY DETECTION STATISTICS. BERRY COUNT – THE NUMBER OF BERRIES REPORTED BY THE ALGORITHM. TRUE POSITIVES – THE NUMBER OF BERRIES DETECTED THAT WERE ACTUAL BERRIES. FALSE POSITIVES – THE NUMBER OF FALSE BERRY DETECTIONS. FALSE POSITIVES – THE NUMBER OF BERRIES VISIBLE IN THE IMAGE THAT WERE NOT DETECTED. RECALL – PERCENTAGE OF VISIBLE BERRIES DETECTED. PRECISION – PERCENTAGE OF DETECTIONS THAT WERE BERRIES.

Variety	Berry Count	True Positives	False Positives	False Negatives	Recall	Precision
Gerwurztraminer	1073	1055	18	354	74.9%	98.3%
Traminette	1116	1096	20	658	62.8%	98.2%
Riesling	784	762	22	657	53.7%	97.2%
Overall	2973	2913	60	1659	63.7%	98.0%

IV. RESULTS

A. Datasets

The results generated in this paper are from three different grape varieties – Gewurztraminer, Traminette and Riesling.

The Gerwurztraminer dataset was collected just before véraison, before color development, and the berries were green in color, see Fig. 1. The Gerwurztraminer dataset was collected from a commercial vineyard and therefore we did not have access to the harvest crop weights. Only 5 vines were included in the dataset and we used it purely for developing the berry detection algorithm.

The Riesling and Traminette datasets were collected from an approximately one acre plot of these *Vitis vinifera* varieties. The Riesling cultivar is a ‘White Riesling’ *Vitis vinifera* and the Traminette is an intraspecific hybrid. We used four rows of Traminette vines and four rows of Riesling varieties, 224 vines in total. The Traminette were at 8ft spacing and Riesling were at 6ft spacing, which totaled 450m of vines.

The vines in this acre plot were vertically shoot positioned and basal leaf removal was performed in the cluster zone, a practice performed by vineyard owners to expose the fruit to the sun to change the flavor characteristics of the grapes. The basal leaf removal also makes yield estimation feasible towards the end of the growing season because the occluding canopy is removed from the fruit-zone. On the Traminette vines the basal leaf removal was performed just on the East facing side of the row and on both sides of the Riesling vines. Our tests captured images from the East side of the rows. Despite not all of the crop being visible from the one side, we calibrate our measurements from a portion of the harvest data, which takes into account the percentage of the grapes that were not visible.

The Traminette and Riesling vines are white grape varieties, the images of the crop were collected post-véraison, and even at this late stage the fruit still had similar coloring to the background of leaves. The similarly colored fruit and leaves demonstrating the ability of our shape and texture approach to detect the crop amongst the canopy.

For our experiments we use a Canon SX200IS, mounted facing sideways at the same height of the fruit zone, capturing images of the crop. The camera is set in continuous capture mode, recording images at 3264 x 2448 resolution, at approximately 0.8Hz. We mount halogen lamps facing sideways, illuminating the field of view of the camera to

improve the lighting of the fruit-zone, which is often in the dark shadows of the canopy. The camera vehicle is driven along the rows in the vineyard capturing images at approximately 0.5m/s.

B. Berry Detection Performance

We first evaluate the performance of our berry detection algorithm, by selecting five images from each of the three different datasets; Gerwurztraminer, Traminette and Riesling. We processed the images with the berry detection algorithm and also manually counted detection statistics, presenting these results in Table I. The shows that our algorithm mistakenly detects only a minimal number of false berries, giving it a very high precision rate. However, it is conservative, it does not detect a sizeable percentage of berries that are visible in the images and therefore has a high false negative count and therefore a moderate recall rate.

To gain an understanding of what part of the algorithm are most responsible for the false negatives detections we break-down the false negatives into the three stages of the algorithm; False detections that are not detected by the radial-symmetry detector (Section III-A), those that are misclassified (Section III-B) and those that are not clustered to neighboring berries (Section III-C). Table II presents the false negative break-down by algorithm stage. The table shows that around 60% of all missed detections are caused by the radial symmetry transform, around 30% are classified as *non-berry* and only 10% of the false negatives are to be blamed on the clustering. We show in the following section that, even with these false negatives, we can still acquire accurate yield prediction because of the high precision rate. However, to further improve performance we could look at modifying the radial symmetry transform to improve the number of berries it can detect without drastically increasing the false detections.

TABLE II
BREAK-DOWN OF FALSE NEGATIVES

Variety	Not-detected	Mis-classified	Not-clustered
Gerwurztraminer	51.7%	31.9%	16.4%
Traminette	73.9%	16.0%	10.0%
Riesling	53.9%	40.2%	5.9%
Overall	61.1%	29.0%	9.7%

C. Multiple Flashes for Improved Interest Point Detection

We have begun investigating a method that can help improve the recall performance of the radial symmetry transform, which was originally proposed by Raskar et al. [6]. The approach is to use multiple flashes placed around the camera to detect the depth discontinuities in the image. The method finds edges that correspond only to depth discontinuities and ignores edges due to texture, Fig. 3(b).

The edge pixels found by an image gradient operator can be due to actual depth discontinuities in the scene (which we are interested in because they could be the contours of grapes) or they can be due to texture in the scene (which can cause the radially symmetry detector to fire at places that are not of interest), Fig. 3(a). Many points are detected with radial symmetry when using the image edges as input, see Fig. 3(c). Whereas if we use the depth edges as input and replace the traditional image gradient, g , in Equation 1, we can isolate the grape berries using the radial symmetry transform, see Fig. 3(d).

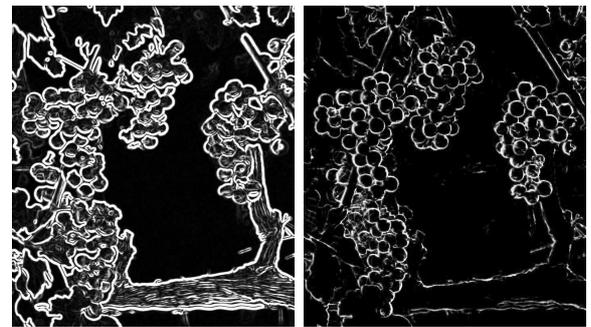
From initial results it is obvious the method of Raskar et al. [6], demonstrated in Fig 3, will improve the performance of our algorithm, however, at the present moment we have not had the chance to deploy the multiple lighting setup on a large scale in a vineyard. Therefore, the yield estimation results presented in the following section use conventional image edges.

D. Yield Estimation

For the yield estimation results, we compare our berry counts against actual harvest weights collected from the Traminette and the Riesling datasets. First, we register images together, and assign registered images to specific vines by defining the boundaries of the vines within the images, cropping-out overlapping content to avoid double counting. We conduct this process manually, but this could be performed automatically if we had in place a localization system, such as GPS and odometry system, which would be able to register data based on the fixed spacing of the vines. See Fig. 4 for examples of our automated berry counts being compared to the harvest data, the row and vine number, the harvest crop weight, and the detected berry count are displayed over the images. Cluster counts are also displayed, however our automated cluster counts were inaccurate because of the difficulties determining separate clusters – late in the season clusters tend to grow over each other. We focus on the berry counts in this work because they produce more accurate yield estimates.

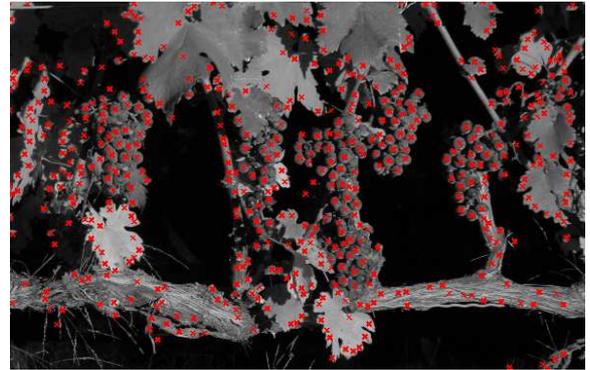
Once registered to specific vines, we compare our automated berry counts with the harvest crop weights. Our automatically generated berry counts produced a linear relationship with actual harvest crop weights with correlation score $r^2 = 0.74$. Fig. 5 shows the datapoints in the correlation and the distribution of measurements.

We saw in Table I that our recall rate is not high and we also know that occlusions will cause further berries to not be counted by our algorithm – yet despite these issues we still get good correlation to the harvest weights. Reasons that our

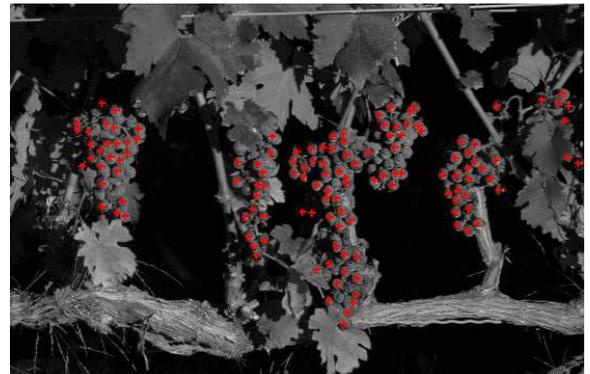


(a) Image-edges

(b) Depth-edges



(c) Radial Symmetry Detection from Image-edges



(d) Radial Symmetry Detection from Depth-edges

Fig. 3. Using the multiple flash method of Raskar et al. [6] to improve performance of radial symmetry transform. Multiple flashes are placed around the camera and triggered sequentially. (a) Regular image processing to find image edges/gradients finds contours in the objects' texture as well as object boundaries. (b) Using the method of Raskar et al. [6] to compute the depth-edges, the boundaries of the objects are isolated, omitting the objects' texture. (c) Radial symmetry detected in regular image edges. (d) Radial symmetry detected in depth discontinuity edges. It is obvious the performance of the radial symmetry detection of grape berries is much improved using the depth edge approach.

measurements achieve good correlation are first through the high precision of our detection algorithm which rarely counts false positives and also because the occlusion level and the percentage of visible berries that are missed has reasonable constancy across the vineyard. Further improvements to the detection algorithm, such as proposed in Section IV-C, and incorporating an estimate of any variations there may be in occlusion will only improve the correlation score.



(a) Traminette



(b) Riesling

Fig. 4. Example showing berry detections for the Traminette and Riesling varieties used in the yield estimation experiment. Detected berries are highlighted by a red contour. The row and vine number, the harvest crop weight, the cluster counts and the berry count are displayed over the images.

Finally, we evaluate the accuracy of our estimates in terms of predicting harvest weight. We fit a function to a part of our dataset that provides a mapping from berry count to harvest weight, and calibrates for the berries that are out of view and missed by the detection algorithm. We calibrate the function using two rows of data (either 48 vines for Traminette or 64 vines for Riesling), and apply the function to the other rows' berry counts.

Once we have functions calibrated from portions of our data we evaluate how accurate our berry counts are at predicting the total weight of other rows of vines for which we have not calibrated our measurements. Fig. 6 presents a graph of the predicted versus actual harvest weights for four rows of Traminette and four rows of Riesling vines. The average error of these results is at 9.8% of the eventual actual harvest weight. An estimate of harvest yield generated taken from measurements at every single vine and achieving 9.8% accuracy for a row, already exceeds what is possible with

current practices that are restricted to very coarse sampling across a vineyard.

V. CONCLUSION AND FUTURE WORK

We have demonstrated that a computer vision method can provide high resolution automated crop yield estimates for vineyard management. Our approach detects and counts crop in images collected from a camera mounted sideways on a vehicle driven along the rows in a vineyard. We combine the traditional crop yield measurements of clusters per vine and berries per cluster, with a single estimate of berries per vine. The number of berries on a vine is known to account for 90% of the variation in harvest yield. We develop an algorithm to detect individual berries in camera images and evaluate in actual vineyard conditions. Unlike other image detection approaches, our approach is not reliant of color contrast, and can detect berries of all colors, even those that are similarly colored to the background of leaves.

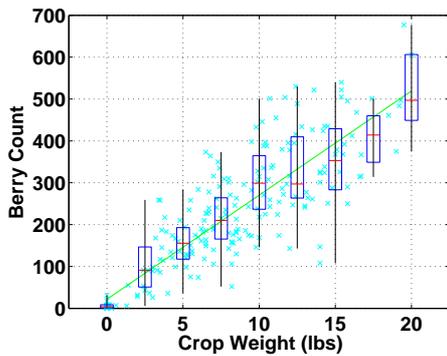


Fig. 5. Correlation between our detected berry count and harvest crop weights gives a correlation score of $r^2 = 0.74$. The box-plot marks show the distribution within the measurements, the green line represents a linear fit and each of the blue data points represents the measurement of a vine, for a total of 224 vines. By comparison, the typical yield prediction approach would take a measurement at a small fraction of the vines and extrapolate, whereas we can measure every vine.

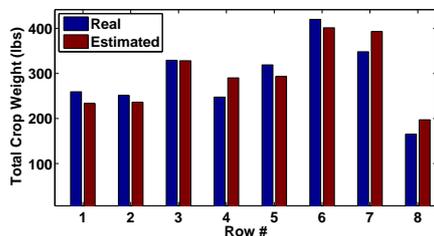


Fig. 6. Graph showing our predictions of the harvest weight of rows in a vineyard. Rows 1 to 4 have 24 Traminette vines each. Rows 5 to 8 have 32 Riesling vines each. Predictions are generated from the functions mapping berry count to crop weight that were calibrated on data from other rows. Our yield estimates have a mean error of 9.8% of the weight of the row. Producing yield predictions at this accuracy at the resolution of single row surpasses the coarse sampling approaches currently used in vineyards.

We evaluate our approach on what we think is the largest automated crop imaging experiment demonstrated in a vineyard. On approximately 450m of vines we compute an automated estimate of the harvest yield using measurements taken from imagery and compare against the actual yield, meticulously measured by hand at harvest time. We compare our measurements to yield and show we can estimate the weight of a row of vines with 9.8% error.

Our results have significance on the future of vineyard operations through our ability to make yield predictions with high fidelity opening up the possibility of vineyard owners making precise adjustments to their vines, where previously they have been restricted to using coarse measurements.

There are a number of avenues of work to further improve

our approach. First is to find ways to improve the recall rate of the current berry detection system, for example by deploying the multiple flash technique discussed in this paper. Another extension would be to augment the berry counts with a method that measures berry size, which is known to account for the remaining 10% of the variation in final yield. In other ongoing work we hope to evaluate how much the function correlating visible berry counts to yield varies by variety, by trellis structure, by differing times of the growing season, and from year to year. We also will look to develop an approach to count grape clusters early in the season, even before berries have formed, to give vineyard managers information with maximum time before harvest to make the necessary adjustments to their vines.

VI. ACKNOWLEDGEMENTS

This work is funded by the National Grape and Wine Initiative, 1415 L Street, Suite 460, Sacramento, California, 95814, USA, info@NGWL.org.

REFERENCES

- [1] Blom, P.E., Tarara, J.M. Trellis tension monitoring improves yield estimation in vineyards. *HortScience*. 44:678-685. (2009)
- [2] Clingeffer, P., Dunn, G.M., Krstic, M. and Martin, S. Crop development, crop estimation and crop control to secure quality and production of major wine grape varieties: A national approach. Grape and Wine Research and Development Corporation, Australia (2001).
- [3] Dunn, G.M. and Martin, S.R. Yield prediction from digital image analysis: A technique with potential for vineyard assessments prior to harvest. *Australian Journal of Grape and Wine Research* 10: 196-198. (2004)
- [4] Federici, J.F., Wample, R.L., Rodriguez, D. and Mukherjee, S. Application of terahertz Gouy phase shift from curved surfaces for estimation of crop yield. In *Applied Optics*, Vol. 48, Issue 7, pp. 1382-1388 (2009)
- [5] Jimenez, A.R., Ceres, R and Pons, J.L. A Survey of Computer Vision Methods for Locating Fruit on Trees, *Transaction of the ASAE*, Vol. 43(6): 1911-1920. (2000)
- [6] Raskar, R., Yu, J. and Ilie, A. A non-photorealistic camera: Detecting silhouettes with multi-flash. In: *ACM SIGGRAPH Technical Sketch*. (2003)
- [7] Loy, G. and Zelinsky, A. Fast Radial Symmetry for Detecting Points of Interest. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 25 Issue 8. (2003)
- [8] Martinez-Casasnovas J.A., Bordes X., *Viticultura de precisin: prediccin de cosecha a partir de variables del cultivo e ndices de vegetacin*. *Revista de la Asociacin Espaola de Teledeteccin* 24, 67-71. (2005)
- [9] Singh, S., Bergerman, M., Cannons, J., Grocholsky, B., Hamner, B., Holguin, G., Hull, L., Jones, V., Kantor, G., Koselka, H., Li, G., Owen, J., Park, J., Shi, W., Teza, J. *Comprehensive Automation for Specialty Crops: Year 1 Results and Lessons Learned*. *Journal of Intelligent Service Robotics, Special Issue on Agricultural Robotics*, vol. 3, no. 4, pp. 245-262. (2010)
- [10] Swanson, M., Dima, C., and Stentz, A.. *A Multi-Modal System For Yield Prediction in Citrus Trees*. *ASABE Annual International Meeting*, Pittsburgh, PA, June. (2010)
- [11] Wolpert, J. A. and Vilas, E. P. *Estimating Vineyard Yields: Introduction to a Simple, Two-Step Method*. *American Journal of Enology and Viticulture* 43: 384-388. (1992)