Article

Predicting Fruit Yield Using Shallow Neural Networks

Trevor Braddock [1], Duke M Bulanon [1,*], Brice Allen and Joseph J Bulanon [2]

[1] Department of Physics and Engineering, Northwest Nazarene University, Nampa, ID USA
[2] Department of Math and Computer Science, Northwest Nazarene University, Nampa, ID USA
* Correspondence: dbulanon@nnu.edu; Tel.: (+1-208-467-8047)

Received: date; Accepted: date; Published: date

Abstract: Precision agriculture is a technology used by farmers to help food sustainability amidst growing population. One of the tools of precision agriculture is yield monitoring, which helps a farmer manage his production. Yield monitoring is usually done during harvest, however it could also be done early in the growing season. Early prediction of yield, specifically for fruit trees, aids the farmer in the marketing of their product and assists in managing production logistics such as labor requirement and storage needs. In this study, a machine vision system is developed to estimate fruit yield early in the season. The machine vision system uses a color camera to capture images of fruit trees during the full bloom period. An image segmentation algorithm based on an artificial neural network was developed to recognize and count the blossoms on the tree. The artificial neural network segmentation algorithm uses color information and position as input. The resulting correlation between the blossom count and the actual number of fruits on the tree shows the potential of this method to be used for early prediction of fruit yield.

Keywords: Artificial neural network; image processing; machine vision; yield monitoring

1. Introduction

One of the issues that governments around the world is facing is food sustainability. With the growing population, food security is a problem that needs to be addressed. It is estimated that in order to feed the world population in 2050, the current crop production needs to be doubled [1]. Currently, farmers are using a technology called precision agriculture that can support food sustainability [2]. Precision agriculture is a site-specific management technique with the goal of maximizing the output while minimizing input [3]. Precision agriculture technologies is very common in row crops. These technologies utilize Global Positioning System, sensors, and variable rate controllers which help farmers save time working their fields and optimizes the use of resources like fuel, chemicals, labor, and fertilizer [4].

One component of precision agriculture is yield monitoring [5], the process of determining how much crop has been produced by the field. Row crop growers over the years have used yield monitoring systems to identify regions of their field that are not performing well [6]. The majority of the current methods developed focus on weighing the crops after they have been harvested [7]. This technique is very effective for row crops such as corn and soybeans. Yield monitors have been studied for specialty crops [8-10], however, expanding this method to specialty crops [11] presents a challenge because of the geometrical parameters of fruits and vegetables. Current methods focus on post-harvest yield which is beneficial but of greater benefit would be a pre-harvest prediction of yield. Fruit farmers, for example, with an early yield estimate, would be able to increase their profitability by achieving an early market competitive pricing and could plan the harvesting and post-harvesting logistics in advance, specifically harvesting labor [12].

Currently, fruit growers estimate their yield by counting fruits during the early fruit drop. They select several trees to count, take the average, and estimate the fruit yield. This method is time consuming and inaccurate. Some researchers have reported the use of machine vision for monitoring yield of specialty crops [13-16]. The Robotics Vision lab of Northwest Nazarene University has developed a method of estimating fruit yield by counting blossoms [17]. It has been reported that there is an increase of photosynthetic activity during the blossom period, which correlates with the fruiting process [18]. Figure 1 shows the correlation of the blossoms detected in an image with the actual number of fruits in the tree. By using this correlation, we were able to estimate the fruit yield by counting the blossoms.

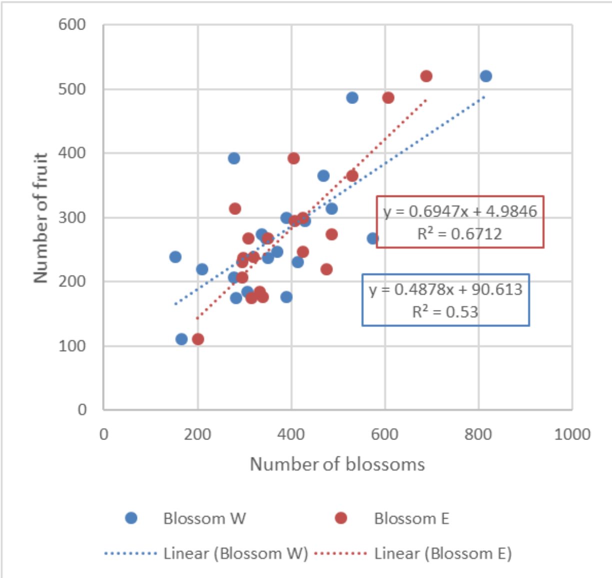


Figure 1. Correlation between number of fruits and blossoms detected from images.

Figure 2 shows the estimated number of fruits versus the actual number of fruits. The segmentation method used in the blossom detection used a linear discriminant function [19]. Although this method proved to be effective, the derivation of the discriminate function is dependent on the data that is used.

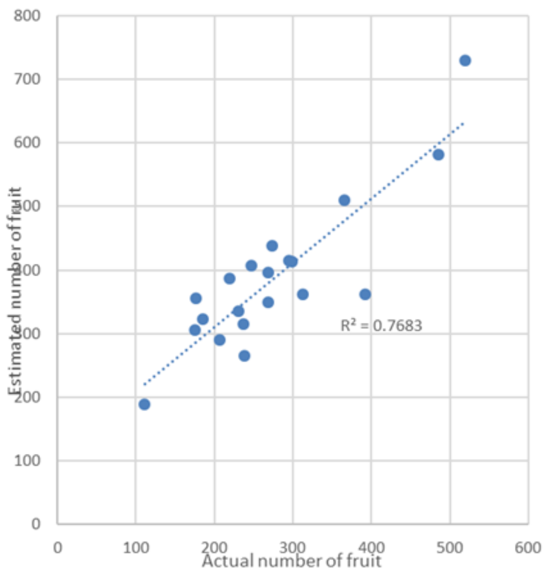


Figure 2. Correlation between actual number of fruits and estimated number of fruits.

The goal of this paper is to improve on the yield monitoring system, specifically on the blossom segmentation process. The objectives of this study are:

1. To develop a shallow neural network, to detect and count blossoms on trees,
2. To evaluate the detection performance of the shallow neural network,
3. To estimate the fruit yield using the shallow neural network.

2. Materials and Methods

2.1. Target Fruit

The target fruit for this study is the Pink Lady Apple from a commercial orchard, located in Caldwell, Idaho. In 2018, thirty trees were randomly selected for this study and no criteria were used for the selection. While in 2019, only 26 trees were used because of uncertain data records. Images of these trees were collected during the full blossom period and then the fruit for the selected trees were counted by hand utilizing a sticker method on individual fruit to avoid double counting in 2018, and a visual count using a hand clicker in 2019 due to stickers potentially causing damage to the apples.

2.2. Image Acquisition

Images of the apple trees were captured using a Canon digital camera (model of the camera) on automatic settings at a height of five feet and a distance of eight feet from the trunk of the tree. Images were taken directly in front of each tree on both the east and west sides. The images of the apple trees had a resolution of 4000x3000 pixels, with 2018 pictures in landscape orientation and 2019 in portrait. They were taken in the morning on April 24, of both years while the sunlight created favorable lighting on the east side of the trees and left the west side in partial shadow. The apple trees were in bloom with most of the flowers in blossom, but with some unfolded blossom buds. Blossoms appeared white in color while the buds were a pink or red color.

2.3. Image Processing and Analysis

Once the photos of the trees had been acquired, image processing [20] was used to count the blossoms on each tree. There were five steps used in the course of image processing and analysis: image cropping, image segmentation, morphological processing, blob analysis, and blossom bounding and count. Images from each step are shown below.

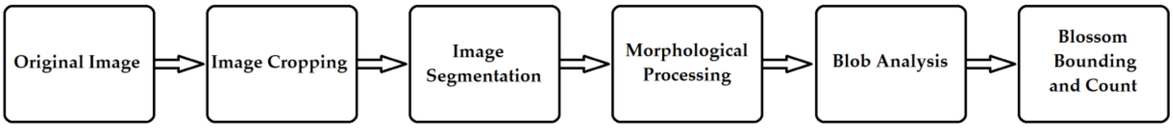
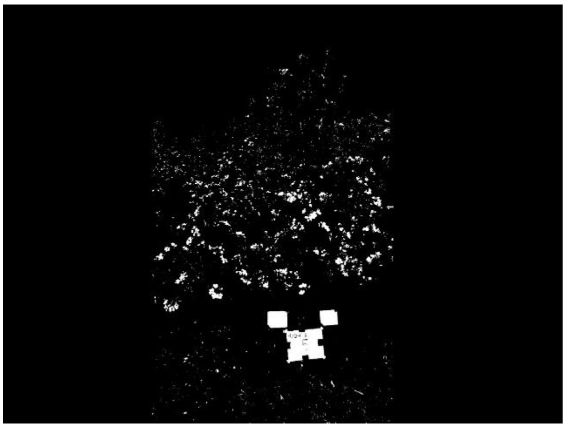
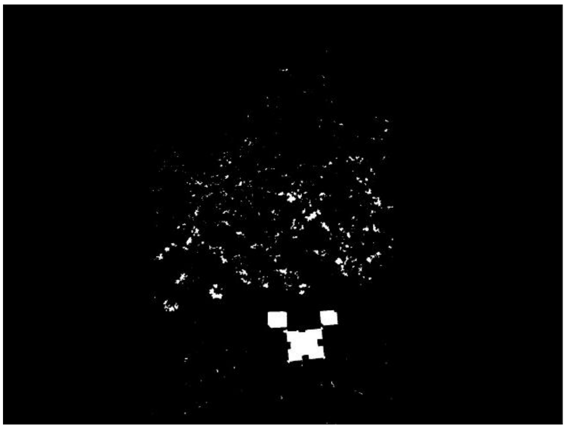


Figure 3. Flow chart of the image processing and analysis process.

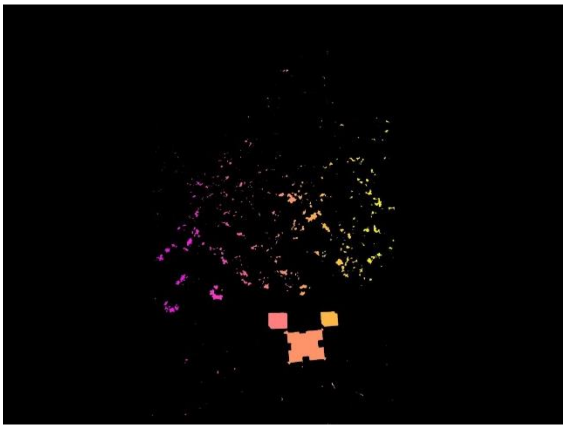
(a) Original Image **(b)** Cropped Image



(c) Segmented Image



(d) Morphological Processing



(e) Blob Analysis

(f) Bounded Blossom Count

Figure 4. Image processing and analysis steps. **(a)** shows the unedited image taken of a blossoming apple tree; **(b)** Shows how the image is cropped to focus on the desired tree; **(c)** Shows the segmented image that is created by the shallow neural network scores for each pixel; **(d)** Shows the segmented image after morphological processing to get rid of the excess noise in the image; **(e)** Shows a representation of blob analysis where each blob is assigned a slightly different color; **(f)** Shows the bounding boxes overlaid on the original image highlighting the detected blossoms, with the count given at the top if the image.

2.3.1. Image Cropping

Image cropping was utilized to focus evaluation on the desired tree because images taken at the orchard can include multiple trees in the same row. First, the image was displayed and two points were chosen on the image to create a left and right cut off. Then any pixels that were located to the left of the left cut off point, with respect to the x-axis of the picture, had their RGB values set to zero, and any pixels to the right of the right cut off point also had their RGB values set to zero. This was done so that only the part of the image between the two selected points, with respect to the x-axis, retained its color value information, while other parts of the image were converted to black. The result of image cropping can be seen in Figure 2(b).

2.3.2. Image Segmentation

Image segmentation is used to separate the blossoms out from the rest of the picture so that they can be evaluated. A shallow neural network was developed for segmenting the image. Two networks were created, one trained on 2018 blossoming apple tree images and the other on 2019 images. The shallow neural networks used to create the binary image consist of three layers, the input, hidden, and output layers. Data is fed into the input layer and then run through the hidden layer which is made up of neurons that run the data through weights and biases to determine the output. The output

 doi:10.20944/preprints202009.0022.v1

is then passed to the output layer and on to the user. The structure of the shallow neural network is shown in Figure 5 below. The shallow neural network used in this paper was trained using the nprtool in MATLAB (2018). Training requires input data and target data. Input data is the same sort of data that the network will be analyzing and consisted of the RGB value of a pixel as well as its X and Y coordinate in the image. Target data is the definition of the input data and is the sort of data that the network will return after analyzing input data. In the case of training the networks, it was either a zero for background pixels or a one for blossom pixels.

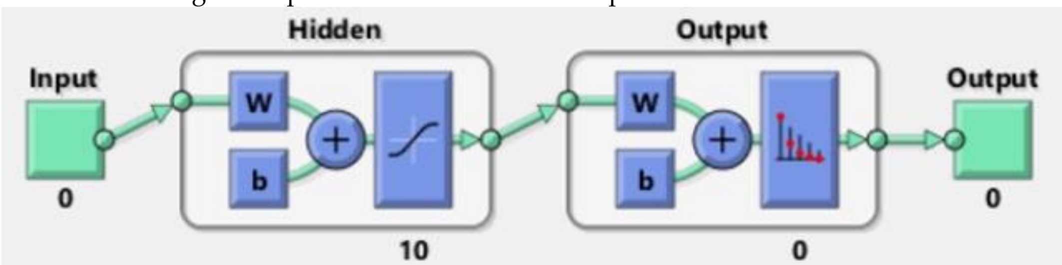


Figure 5. Structure of the shallow neural networks, where pixel data is passed into the input layer and a confidence score is passed out of the output layer.

For training the shallow networks, a program was created in MATLAB that allowed a user to select pixels in an image while designating them as either background or blossom pixels. The program would then gather all the input data from the pixels as well as the target data designation and write it to a spreadsheet file that was used for training the shallow neural networks. The trained network is used for image segmentation.

To segment the image, a set of data was created for each pixel in the image, containing the color and position data of the pixel. After the data was collected from the image, each pixel data set was processed by the shallow neural network, returning a score of the likelihood that the pixel was part of a blossom. The shallow neural network returned scores that ranged between zero (low match) and one (high match). A binary, segmented image was created from the scores by using a threshold of 0.75; scores >0.75=1 and scores ≤0.75=0. This resulted in a binary image where pixels with a value of one are white and pixels with a value of zero are black, as seen in Figure 2(c).

2.3.3. Morphological Processing

Morphological processing was used to eliminate the noise from the segmented image and retain only the blossoms. Three MATLAB functions were used in the morphological processing to fill holes and erode the edges of blobs of white pixels. The purpose of erosion was to eliminate small groups of white pixels and groups that did not have the build of a blossom, and also to separate blossoms that had a small overlap. While filling holes was used to prevent erosion from destroying larger groups of white pixels. These functions were used repetitively to achieve as much noise reduction as possible while still maintaining most of the true blossom pixels. The result of morphological processing can be seen in Figure 2(d).

Blob Analysis

Blob analysis is the process of getting the properties of the groups, or blobs, of white pixels that represent potential blossoms in the binary image. The functions were used in MATLAB to label each blob and get their properties, which included area, bounding box position, and centroid position. Figure 2(e) shows each blob as a different color, representing the individual labeling and analysis of each blob.

Blossom Bounding and Count

Blossom bounding is the process of putting a box around the blossoms in the original color image. This is done by using the blob properties from the binary image to get the blossom bounding box position. Not every blob that remained after morphological processing is a blossom, so a final size filter is used when choosing which blob bounding boxes to overlay on the original image. Both blobs that are too small and those that are too big were filtered out and then the remaining blobs were counted and their bounding boxes and centroids overlaid on the original color image, as shown in Figure 2(f).

3. Results

3.1. Blossom Detection

The detected blossoms for each tree versus the hand count of fruit at the end of the season is plotted in the figures below

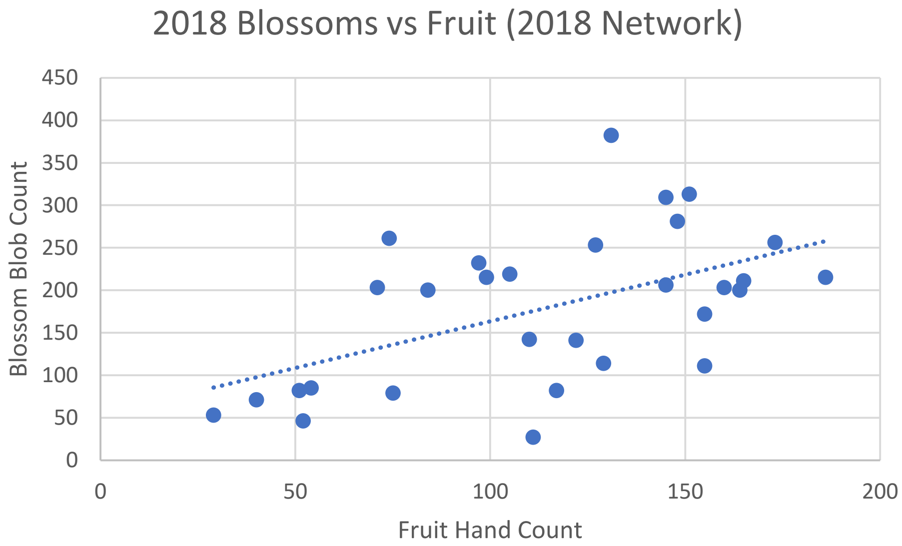


Figure 6. Plot of blossom blob count of 2018 data using 2018 network versus the fruit hand count.

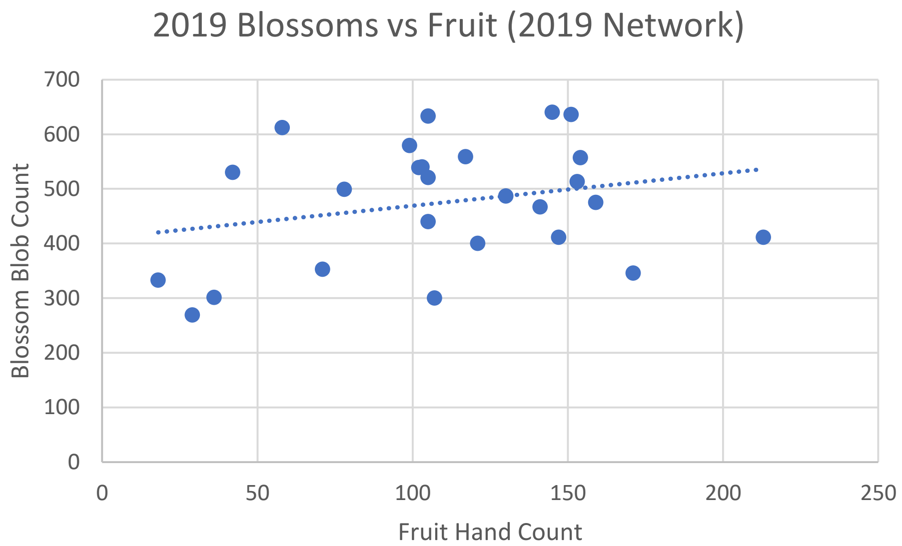


Figure 7. Plot of blossom blob count of 2019 data using 2019 network versus the fruit hand count.

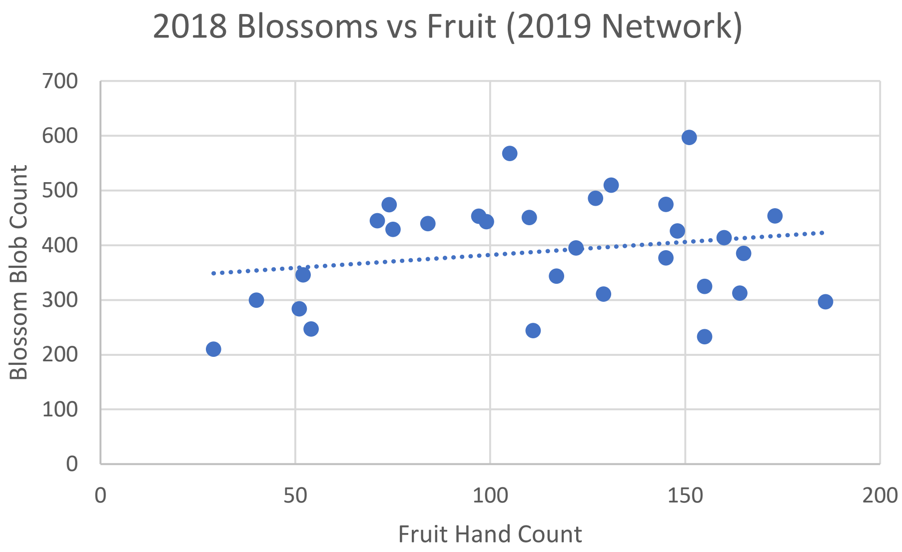


Figure 8. Plot of blossom blob count of 2018 data using 2019 network versus the fruit hand count.

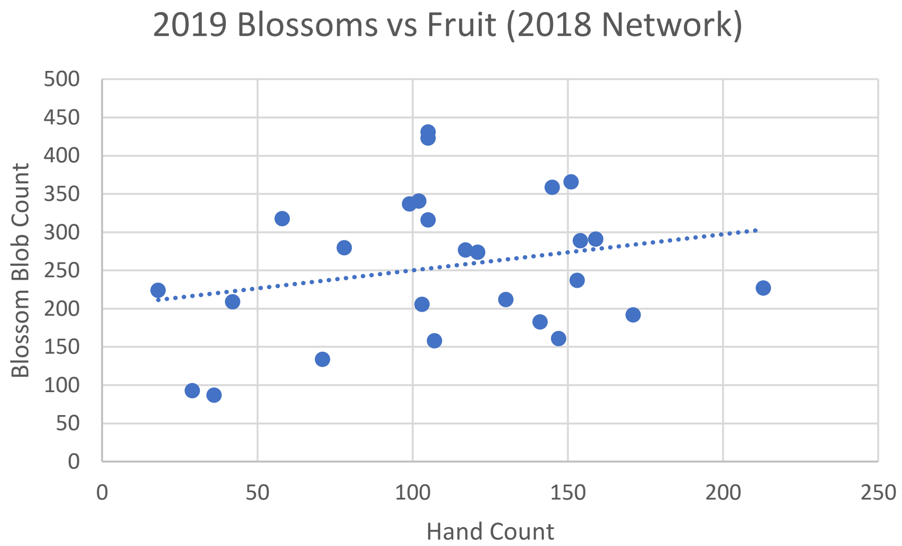


Figure 9. Plot of blossom blob count of 2019 data using 2018 network versus the fruit hand count.

When detecting blossoms, the MATLAB program returns both a blossom blob count and a count of total blossom pixels in all the blobs. This results in six detection results for each tree: the blob count and the pixel count for the east side, west side, and combination of east and west sides. The correlation between the blossom detection results and the fruit count was found for all six parameters for each combination of year's data and network, with the highest scores for each combination given in Table 1. The highest correlation was achieved using the blossom pixel count on the east side except for the 2019 data evaluated using the 2018 network, which had the highest correlation coming from the blossom blob count on the east side.

Table 1. The highest correlation scores between blossom detection results and fruit hand count for all four combinations of year's networks and data.

	Data Year	
	2018	2019

 doi:10.20944/preprints202009.0022.v1

Network Year	2018	0.55	0.24
	2019	0.36	0.32

3.2. Fruit Estimation

To test the ability to accurately predict fruit yield, the trendline equation between blossom detection results and fruit hand count was used as a prediction model to process other detection results. The prediction of 2018 fruit using 2018 network was done by using 2/3 of the data to create the model and the remaining 1/3 to test the model. The same was done for 2019 respectively. The prediction of 2019 fruit using 2018 network was done by using all the 2018 data evaluated with the 2018 network to create the model and testing on all the 2019 data. And vice versa for the prediction of 2018 fruit using 2019 network. The tables below give the detection results and errors.

Table 2. The fruit prediction for 2018 and 2019 using 2018 network.

Data year evaluated using 2018 Network	Total Predicted Fruit Count	Total Actual Fruit Count	Error
2018	1169	1225	-4.6%
2019	2721	2860	-4.9%

Table 3. The fruit prediction for 2018 and 2019 using 2019 network.

Data year evaluated using 2019 Network	Total Predicted Fruit Count	Total Actual Fruit Count	Error
2018	2518	3425	-27%
2019	1003	1075	-6.7%

4. Discussion

The correlation results were poor, with only the 2018 data evaluated using 2018 network having a reasonable correlation of 0.55 as seen in Table 1. However, when looking at prediction capabilities the 2018 network was able to predict total fruit yield with a small amount of error in both years data. The 2019 network was not as capable at making the transition to another year's data, with an error of -27% on the 2018 data as seen in Table 3. These networks may not be capable of successfully predicting on a tree by tree basis, but when looking at a whole orchard they may be able to create a model that can successfully predict total fruit yield early in the season with a small margin of error.

It is important to note that the blossom blobs detected by the program do not consistently represent a single blossom. Rather the blobs tend to be clusters of blossoms. This is due to the regular overlap of blossoms in the photos. In reality there are many times the number of blossoms on the tree than the number of blobs detected. So, the exact relationship between the number of blossoms and the number of harvested apples is not completely demonstrated in this research. A more accurate representation of individual blossom count could potentially be achieved using a size filter to categorize each blob by how many blossoms it could contain. However, this method would be inconsistent depending on the distance from the tree that pictures are captured because blossoms would appear larger the closer to the tree the camera was. A better method may be utilizing an area to circumference ratio to evaluate how many blossoms are in each blob. Although the effectiveness of this method would likely be limited to blobs with few blossoms that only have a small overlap.

5. Conclusions

A machine vision system was developed to estimate apple fruit yield early in the season. The machine vision system was composed of an RGB camera and a personal computer to process the images. To estimate the yield early in the season, images of the trees during the blossom period were acquired and the number of blossoms were correlated with the actual number of fruits on the trees. Two shallow neural networks were developed for blossom detection using an input of the RGB values of a pixel as well as the X and Y coordinate of the pixel in the image. The 2018 shallow network performed better than the 2019 network with average errors of -4.8% and -17% respectively in estimating the number of fruits. This method of estimating fruit yield using blossom detection has potential in helping farmers manage their orchards.

Author Contributions: Duke M. Bulanon conceived the study, designed the experiments, directed the collection of data, and wrote the paper. Brice Allen assisted in the design of experiments, collection, processing, and analyzing the data. Trevor Braddock collected the data, developed the algorithm for image processing, process and analyze the data, and wrote the paper. Joseph Bulanon collected and process the data.

Funding: This research was funded by the Idaho State Department of Agriculture through the Idaho Specialty Crop Block Grant 2018.

Acknowledgments: The authors would like to acknowledge Symms Fruit Ranch and Williamson Orchard for allowing the Robotics Vision Lab of Northwest Nazarene University to use their orchards for the study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Food and Agriculture Organization of the United Nations. 2050: A third more mouths to feed. Available online:http://www.fao.org/news/story/en/item/35571/icode/http://www.fao.org/news/story/en/item/35571/icode/ (accessed on 27th August 2020).
2. Bongiovanni, R; Lowenberg-Deboer, J. Precision Agriculture and Sustainability. Precision Agriculture 2004, 5, 359-387
3. Zhang, Q. Precision Agriculture Technology for Crop Farming; CRC Press Taylor & Francis Group, Boca Raton, FL, USA
4. The Precision Farming Guide for Agriculturists, 4th Edition. 2017. Deere & Company, Moline, Illinois
5. Panda, S.S., Ames, D.P., Panigrahi, S. 2010. Application of Vegetation Indices for Agricultural Crop Yield Prediction Using Neural Network Techniques. Remote Sens. 2010, 2(3), 673-696
6. Magalhães PSG, Cerri DG P. Yield monitoring of sugar cane. Biosystems Engineering 2007, 96 (1), 1–6
7. Fulton, J., E. Hawkins, R. Taylor, A. Franzen 2018. Yield Monitoring and Mapping. In: D.K. Shannon, D.E. Clay, N.R. Kitchen, editors, Precision Agriculture Basics, ASA, CSSA, and SSSA, Madison, WI. p. 63-78. doi:10.2134/precisionagbasics.2016.0089
8. Ampatzidis, Y.G., Whiting, M.D., Liu, B. et al. Precision Agric (2013) 14: 162. https://doi.org/10.1007/s11119-012-9284-3
9. Zaman, Q.U; Schumann, A.W; Hostler, H.K. Estimation of Citrus Fruit Yield Using Ultrasonically-Sensed Tree Size. Applied Engineering in Agriculture. Vol. 22(1): 39-44. (doi: 10.13031/2013.20186) @2006
10. Grift, T; Ehsani, r; Nishiwaki, K; Crespi, C; Min, M. Development of a Yield Monitor for Citrus Fruits 2006 ASAE Annual Meeting, (doi: 10.13031/2013.20661) @2006
11. Lee, W.S., Alchanatis, V. Yang, C., Hirafuji, M., Moshou, D., Li, C. 2010. Sensing technologies for precision specialty crop production. Computers and Electronics in Agriculture. 74(2010), 2- 33.
12. Manfrini, L; Peirpaoli, E; Zibordi, M; Morandi, B; Muzzi, E; Losciale, P; Corelli Grappadelli, L. Monitoring Strategies for Precise Production of High Quality Fruit and Yield in Apple in Emilia-Romagna. Chemical Engineering Transactions 2015, Vol. 44, 2015, 301-306
13. Annamalai, P; Lee, W.S. Citrus Yield Mapping System Using Machine Vision 2003 Paper number 031002, 2003 ASAE Annual Meeting . (doi: 10.13031/2013.13701) @2003

14. Zaman, Q. U; Schumann, A. W; Precival, D. C; Gordon, R. J. Estimation of Wild Blueberry Fruit Yield Using Digital Color Photography. Transactions of the ASABE 2008 51(5), 1539-1544
15. Dorj, U. O; Lee, K; Lee, M. A Computer Vision Algorithm for Tangerine Yield Estimation 2013 International Journal of Bio-Science and Bio-Technology. 5(5), 101-110
16. Wang Q., Nuske S., Bergerman M., Singh S. (2013) Automated Crop Yield Estimation for Apple Orchards. In: Desai J., Dudek G., Khatib O., Kumar V. (eds) Experimental Robotics. Springer Tracts in Advanced Robotics, vol 88. Springer, Heidelberg. https://doi.org/10.1007/978-3-319-00065-7_50
17. Braun, B., Bulanon, D.M., Colwell, J., Stutz, A., Stutz, J., Nogales, C., Hestand, T., Verhage, P., Tracht, T. 2018. A Fruit Yield Prediction Method Using Blossom Detection. 2018 ASABE Annual International Meeting 1801542.(doi:10.13031/aim.201801542)
18. Fujii, J.A.; Kennedy, R.A. Seasonal changes in the photosynthetic rate in apple trees—A comparison between fruiting and nonfruiting trees. Plant Pysiol. 1985, 78, 519–524.
19. Li, B. & Yu, Q. 2008. Classification of functional data: A segmentation approach. Computational Statistics & Data Analysis, 52(10), 4790-4800.
20. Gonzalez, R.C. & Woods, R.E. 2017. Digital Image Processing (4th Edition). Pearson (March 30, 2017).