A Comparison of Computer Vision Techniques for Classifying and Localizing Fruits in Images: Handcrafted Features Vs. Deep Networks

Gautam Pradeep

University of California, Davis
1 Shields Ave, Davis, CA 95616
(916)-293-1685

Email of Author: gautam0831@gmail.com

A Comparison of Computer Vision Techniques for Classifying and Localizing Fruits in Images: Handcrafted Features Vs. Deep Networks

Gautam Pradeep

Mentor: Krishna Kumar Singh
UC Davis Faculty Advisor: Professor Yong Jae Lee

Abstract

Fruit detection is an important initial step to make autonomous robots for various agricultural tasks such as harvesting fruits, analyzing ripeness, and detailed mapping of fields according to fruit density. We can couple a fruit detector with mechanical sensors and components to create a fully-functioning autonomous robot to harvest fruits at a far greater rate.

The objective is to discover the optimal approach for fruit detection from images, a relatively unseen research. We use a machine learning-based method to classify different fruits by not only determining what is the object of interest, but also its location by bounding box prediction. We approach this problem in two ways: 1) Handcrafted Features, 2) Deep Networks. While implementing handcrafted image features such as color histograms, Histogram of Oriented Gradients (HOG)[5] , and GIST [1], we train a logistic regression classifier to learn a fruit detector. We also train a detector under a state-of-the-art method–deep learning–and compare the results on 250 test images with handcrafted learning features, both qualitatively and quantitatively over three trials. We analyze the deep network, which is built upon the You Only Look Once (YOLO) deep learning platform, by changing the amount of training data to observe its effect on detection performance. After experimentation, we found that deep networks with both convolutional and fully-connected layers perform far better than handcrafted features, as the mean average precision (mAP) for the deep network was 45.3% higher than that of the combined handcrafted detector. The mean average precision was calculated from our Precision versus Recall curves for corresponding detectors.

Key Words: Detector, Training, Precision, Recall, Convolutional

1. Introduction

Many agricultural fields today spend millions of dollars on labor and equipment for harvesting fruits, growing the crops, and maintaining infrastructure. Automated robots can revolutionize agricultural fields all around the world by identifying pest and disease breakout before they actually occur, managing crop quality, harvesting fruits without human interaction, and the list goes on. However, most of these fields have still not integrated automated machines to augment production yield [2]. Automated robots have the power to drastically diminish the cost of labor and the time necessary for certain agricultural tasks such as harvesting of fruits. The ability of the robot to recognize a fruit and

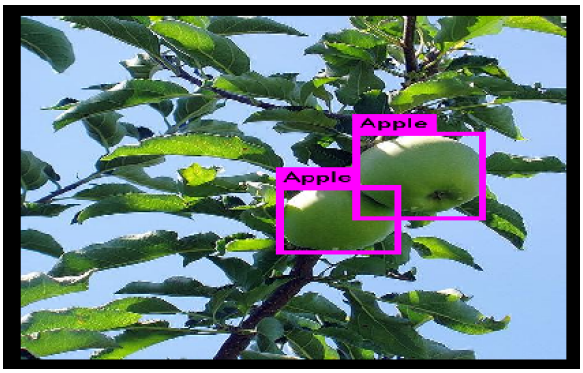


Figure 1: Deep learned YOLO detector: After training the detector on an end-to-end neural network with YOLO [7], the testing output provides the predicted bounding boxes with the highest confidence probability and also labels the class with the highest class-specific probability

localize it is the core of the automated machine since this is the first and most crucial step, which can be learned with machine learning in image classification, as accomplished with our research. This can be done by training a classifier over thousands of different images of fruits.

There are various handcrafted features from which a fruit detector can recognize patterns and then classify and localize different fruits in images. Certain handcrafted features include color histograms, Histogram of Oriented Gradients (HOG [5]), and GIST [1]. Each one of these features captures different properties of the fruits in the image. A color histogram determines the frequency of RGB (color) values for each object in the image. HOG [5] focuses on patterns such as the edges and outlines of the objects. Finally, GIST[1] computes a multitude of quantities such as color distribution in RGB channels, orientation, and edge data. Using a logistic regression [8] classifier allows the program to create hyperplanes to distinguish data on a multi-dimensional plane for each of the classes (types of fruits). This will allow the detector to classify different fruits in images with different confidence values. However, a detector should be able to not only classify the object in the image, but also localize it. This can be done through candidate box generation. Using an algorithm known as Selective Search, many candidate boxes can be proposed for a single object in an image. This allows for selecting positive candidate boxes with high overlap with the *ground truth* box (human-labeled box for object) and also negative candidate boxes (background) with little or no overlap. Both the positive and negative boxes are used during classification.

Deep learning is another learning technique based on end-to-end neural networks. These networks consist of many layers, some convolutional (locally connected) and some dense, fully-connected layers. Using this learning mechanism, the program will learn different features and filters on its own by passing information through a series of neurons at each layer. YOLO [7], a deep learning framework, allows for efficient and accurate object detection in real-time and is implemented with Darknet [6], a framework for deep learning, allowing a platform to initialize the layers and operate the deep network. In deep learning, both the feature and the classifier are learned together as a part of an end-to-end network, whereas for the handcrafted classifier, the features are fixed. Using precision-recall curves and data visualization, quantitative and qualitative results can be examined, respectively to further understand the extent of this research. We compare the results of the deep neural network with the hand-crafted detector quantitatively.

2. Materials and Methods

2.1 Handcrafted Detector

We are given a supervised image collection of one thousand images of five different classes of fruits–apples, mangoes, oranges, peaches, and pears, with two hundred images for each class.

There are three main features which we will be calculating to be learned by the detector. We will compute a color histogram for the images by allocating bins for pixels and appending the bins based on the RGB values of the pixels on a multidimensional plane. Also, we use the Histogram of Oriented Gradients (HOG [5]), a feature descriptor, to count the occurrences of gradient orientation of localized portions of an image. This descriptor implements attributes of edge orientation histograms. The feature analyzes the edges and the outline of the object in an image by converting the image to grayscale. The final feature used in the handcrafted detector is GIST [1], a holistic representation of the spatial envelope. This feature proposes a set of several perceptual dimensions such as texture, openness, expansion, etc. This model is very applicable as it can be used to detect objects based on the scene and the surroundings – the "gist."

Furthermore, we use Selective Search to generate bounding box proposals to then be able to set a threshold and compute the Intersection over Union of the proposal box and the ground truth box. We will discuss the proposal phase of the research later in this paper.

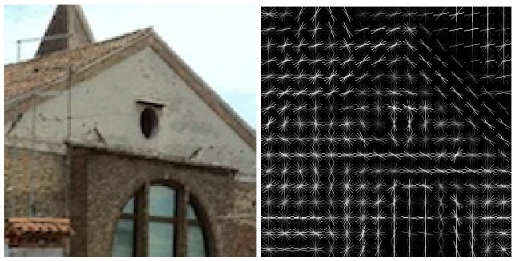


Figure 2: Basic HOG Computation [5]: Displays the standard HOG [5] feature with a cell size of eight pixels. The input image is converted into a gray colormap and is computed with a VLFeat computation

After running the color histogram generation on each ground truth box, extracted from each image, we train a linear classifier to learn from the color histogram data alone. We use a multidimensional logistic regression classifier, using both the positive and negative proposals to create a hyperplane between the data points (the color histogram points) to indicate which class a specific image belongs to. The challenge is to train a machine to understand structure from the data and map it with the correct and true class label. The model classifies the training data by approximating the probability, given the underlying system, of a certain input outputting the desired value for the predicted label as well as the coordinates for the predicted bounding box.

Given an underlying system *S* corresponding to xq, the system's output is yq.

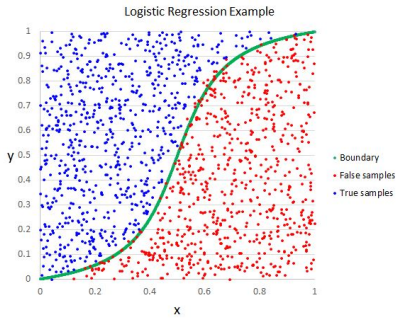


Figure 3a: Sample Logistic Regression Classifier: Distinguishing between 2 classes in this example, but 5 such classes in this research. Source: https://helloacm.com/a-short-introduction-logistic-regression-algorithm/

The logistic regression classifier computes the optimal hyperplane using Lagrange multipliers in mathematical optimization to find local maxima or minima of a function subject to equality constraints. During this project's training, we used a support vector machine from Liblinear [8] with a logistic regression option. We chose the logistic regression option instead of the standard dual multi-class SVM or the linear SVM because the logistic provides probability values more naturally, from zero to one, during testing.

The color histogram was computed by processing not only each image, but also each subpart of the image containing the fruit one by one. Designed to calculate each pixel's intensity of color from a red-green-blue (RGB) scale, the histogram is binned with the value on a multidimensional hyperplane. We normalize each value in the binned histogram through an Euclidean normalization which regards taking the square root of the total sum of all the squares of all the individual values.

The Histogram of Oriented Gradients (HOG [5]) attempts to build feature descriptors. Feature descriptors are representations of an image that extract useful information and throw away extraneous information. HOG [5] first preprocesses the input image and resizes specific extracts to have an aspect ratio of 1:2. Using kernels and filtering the image, HOG calculates the horizontal and vertical gradients simultaneously. The image is then divided into 8x8 cells and a histogram of gradients is calculated for each cell. The histogram is a vector of nine bins corresponding to angles 0, 20, 40, 60, …, 160, and is very informative on the edges and the relative outline of the object of interest.

The holistic representation of the "spatial envelope" is presented by GIST [1] through a series of segmentation and region processing. Again, we normalized the outputs of the GIST feature through Euclidean normalization as used for HOG [5]. GIST [1] approaches the image detection problem with a very generic strategy of localization using the sliding window approach. The gist also captures texture and spatial layout of an image, understanding both the region of interest as well as possibly occluding noise.

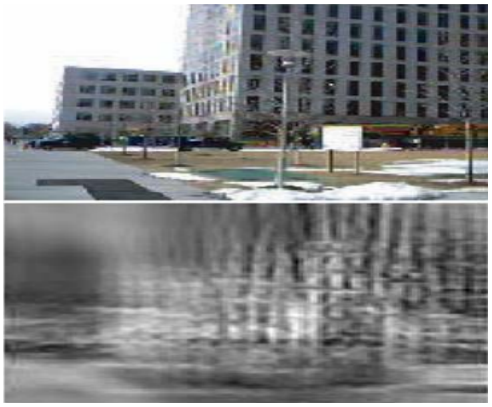


Figure 3b: GIST[1]: Conversion of image during processing

2.2. Deep Network Detector

Deep networks have shown great promise in the field of computer vision on general detection tasks, but we wanted to see how effective a deep feature detector would be on a specific fruit dataset. In most of the related works, deep learning has not been implemented much into the field of fruit detection.

In this research, we use YOLO [7] "You Only Look Once," which is a deep learning architectural platform pretrained on larger general image collections such as ImageNet. These pre-trained weights transfer over when training on our own dataset, which increase the learning capabilities. Even though YOLO [7] had already been created for the ease of future deep detection tasks, we still need to retrain the model to accommodate for our dataset. Using Darknet [6], the deep learning base from which YOLO [7] was based off of, we are able to not only train the model to learn the features and the classifiers in an end-to-end network, but also we are able to obtain natural probability estimates during evaluation, essential for the precision-recall analysis.

After an input image is delivered into the neural network, the early convolutional layers first split the image into a 7 x 7 grid [7].

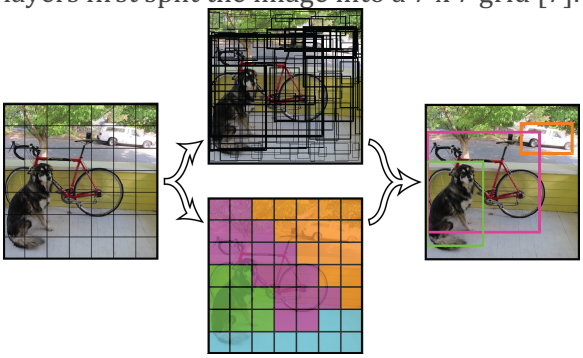


Figure 4

Source:https://github.com/pjreddie/darknet/wiki/YOLO:-Real-Time-Object-Detection

As we progress through the convolutional layers, we see that all predictions for bounding boxes are forward-passed simultaneously. Unlike other deep architectures such as Faster-RCNN [9], all predictions are sent through the network at once using the grid design, which drastically reduces the computation speed with negligible compromises in performance accuracy.

The YOLO [7] deep network consists of 24 convolutional and 2 fully-connected layers. The convolutional layers are locally-connected and capture low level semantics such as color, edges, outline, and spatial orientation. As the network gets more and more complex through the hierarchical representation, the later fully-connected layers are instituted to be able to capture high level semantics such as background and specific segments of the object-of-interest for complex feature extraction. Fully-connected layers are very dense, as each neuron from the previous layer is passing information to each other neuron in the following layer.

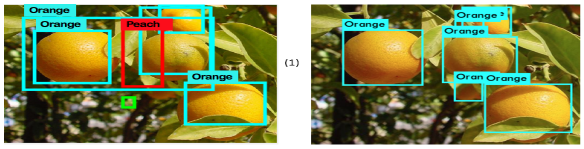
Finally, from each of the forty-nine boxes created from the grid separation technique in YOLO, two bounding boxes are proposed and predicted. After the top two are predicted based on probability and confidence values, the top one box will be chosen from each grid [7]. This is a drawback of YOLO because if there are multiple fruits in a small area of space within one grid, only one box can be drawn for all the fruits in that grid [7]. However, we desire individual bounding boxes for separate fruits matched with correct classifications.

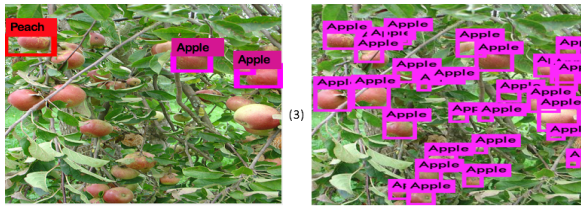
4. Results

Specifically, we analyzed the performance of the handcrafted and the deep detectors both qualitatively and quantitatively. Qualitatively, we present a multitude of images with its corresponding combined handcrafted detections and deep feature detections.A combined handcrafted detector is a concatenated version of the separate feature extraction for the color histogram, HOG [5], and GIST [1] to train a global classifier. Quantitatively, we calculate the average precision of each of the detectors. To find the average precision, we take the area under the curve of the precision versus recall graph. If for example there are ten apples in an image, and five are classified. However, out of these five classifications, only three have been correctly classified. This means our

precision is ⅗, 0.6, and our recall is 3/10, 0.3. After plotting this curve, we are able to assess the performance of the detector by calculating the mean average precision, which is the average precision over all five classes.

Handcrafted **Deep**





4.1 Quantitative Results of Precision-Recall

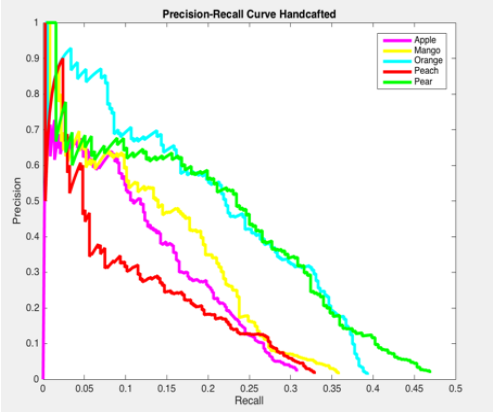
Mean Average Precision over 5 Classes (Combined Handcrafted Detector/Deep Detector)					
Trial	Apple	Mango	Orange	Peach	Pear
1	0.1144 **/0.5389**	0.1354 **/0.6938**	0.2071 **/0.5281**	0.0896 **/0.5627**	0.1988 **/0.7381**
2	0.1158 **/0.5367**	0.1285 **/0.6811**	0.2036 **/0.5748**	0.0890 **/0.4956**	0.1890/ **0.6712**
3	0.1202 **/0.5115**	0.1236 **/0.6508**	0.2160 **/0.5774**	0.0879 **/0.5381**	0.1829 **/0.7036**

Area Under Curve of Precision-Recall for Handcrafted and Deep					
	Color	HOG	GIST	Combined HF (avg)	Deep (avg)
Apple	0.0477	0.0595	0.0430	0.1168	**0.5290**
Mango	0.0098	0.0390	0.0387	0.1292	**0.6752**
Orange	0.1177	0.0517	0.0419	0.2089	**0.5601**
Peach	0.0341	0.0689	0.0421	0.0889	**0.5321**
Pear	0.0293	0.1030	0.0784	0.1902	**0.7022**

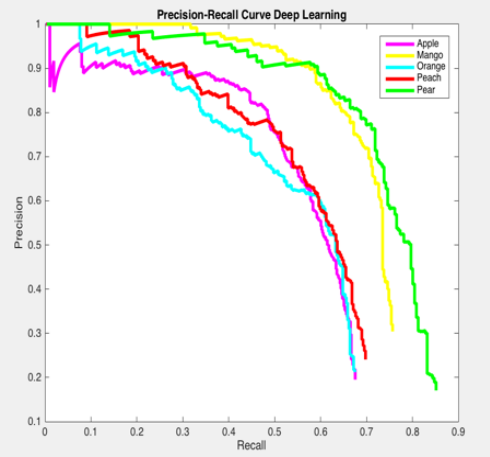
Mean Average Precision (mAP) over Five Classes				
Color	HOG	GIST	Combined HF	Deep
0.0477	0.0644	0.0488	0.1468	0.5997

Mean Average Precision Vs. Amount for Training Data Per Class for Deep Feature		
50 images	100 images	150 images
0.2619	0.5934	0.6111

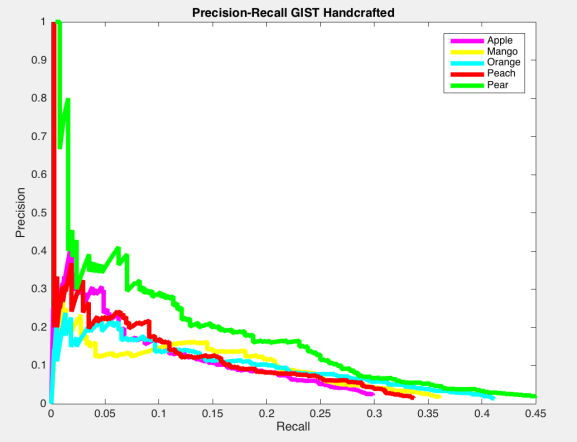
Handcrafted Combined Results



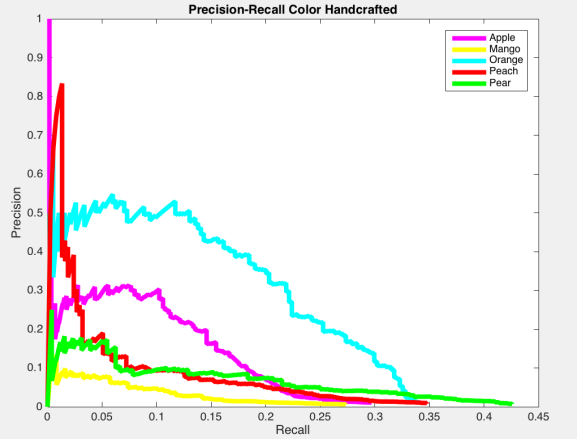
Deep Network Results



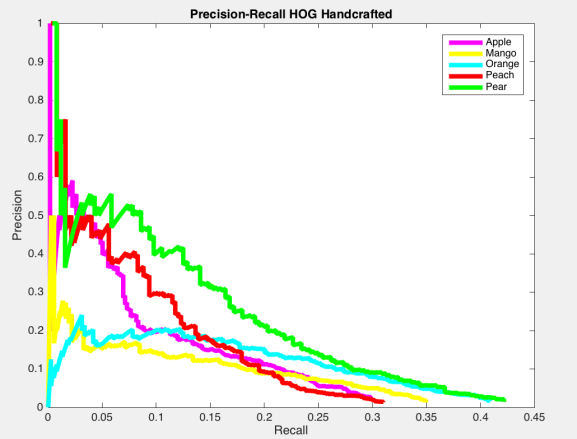
Individual GIST Results



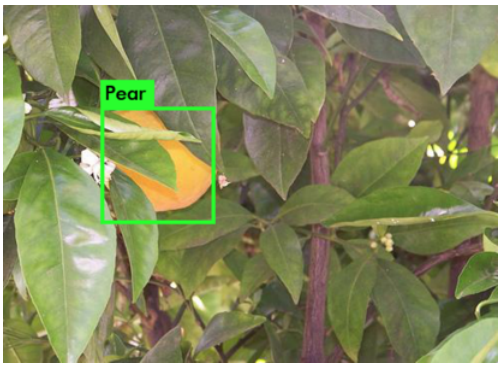
Individual Color Results

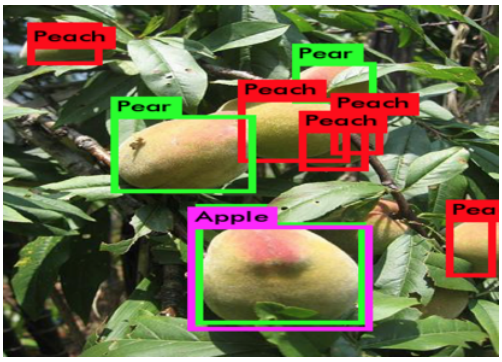


Individual HOG Results



4.2 Failure Cases for the Deep Detector





5. Discussion

Qualitative Analysis

Based on the **qualitative data**, one can see that the deep features are able to accurately classify and localize the fruits in the images far better than handcrafted features. In

Image 3 we see that the deep feature is able to classify and locate a much higher percentage of the apples present, and also does not have any misclassifications.

The handcrafted feature may have failed here due to the small size of the fruits themselves, as well as possible occlusion from the leaves and background. Due to low capacity, handcrafted features cannot perform as well as deep ones for challenging scenarios mentioned above. From the **qualitative data** we can also see why oranges are very easily classified in both the handcrafted and the deep detector due to its unique color, compensating for its easily-confused spherical shape.

From the **failure cases of the deep detector,** it is clear that occlusion and cluttered backgrounds make detection very challenging.

Some ways failure cases could arrive is from light variation, the amount of fruits, the size of the fruits, and any deformities in the outline/shape. Long-range images sometimes result in bounding boxes that enclose multiple fruits. Another failure case includes leaves on trees being classified as pears or apples due to color and also the viewing angle.

Quantitative Analysis

The area under the curve of a precision-recall curve represents average precision. The greater the average precision, the better a detector performs on that specific class. Clearly, the HOG [5], GIST [1], and color histogram perform the best when combined since they provide **complementary information**.

For each class, the deep feature performed better than the handcrafted features. Deep feature for the mango and pear classes gave a 0.5460 and 0.5992 increase in the average precision, respectively, compared to the color feature on the same test images. However, even after the concatenation, **the deep feature increased the average precision across all fruit classes by 0.4529 in comparison to the combined handcrafted feature.** Fruits with areas less than 400 pixels are not considered in evaluation, as they are very hard to identify visually

With more training data samples, there is a direct correlation with a higher mean average precision. The end-to-end neural network consists of many layers of nonlinear processing to form a hierarchical representation of feature extraction, making it more accurate than handcrafted counterparts.

6. Conclusion

We can see that the deep detector performed far better than the handcrafted feature detector. An extension to this project would be to train the deep detector with more training data samples across a wider variety of fruit classes. Also, training from videos instead of images may also have a beneficial impact to the detector's performance. This can lead to various applications in the future using the deep detector.

There are many applications this project poses in the real-world. Implementing this detector in robots is the first and most important step in creating an autonomous device to harvest these fruits mechanically. Using this detector in agricultural fields can drastically decrease cost of labor and increase harvest yield.

A camera for visual sensing, coupled with a detector to classify and localize the fruit can be joined with a mechanical sensor to calculate the distance between the robot and the fruit tree which will then trigger an arm to pluck the fruit from the tree.

The world population is expected to rise to 9 Billion by 2050, and this means that farmers will have to produce 70% more food than ever before [3]. Using automated fruit detectors with visual detection systems such as the deep detector created in this project, this is definitely achievable. With the implementation of the fruit detector in autonomous robots, the cost of labor will drastically decrease and the harvest yield will be boosted substantially. Furthermore, any automation will also leave less of a human-mark on nature and has the potential to be far more environmentally friendly by consuming less energy.

7. Acknowledgements

I acknowledge Krishna Kumar Singh, my mentor, and Professor Yong Jae Lee for their continuous support and appropriate guidance throughout this extensive research.

8. References

[1] A. Oliva and A. Torralba. Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope. In International Journal of Computer Vision, 2001.

[2] Bac C.W., Hemming J., van Henten E.J. Robust pixel-based classification of obstacles for robotic harvesting of sweet-pepper. Comput. Electron. Agric. 2013;96:148–162. doi: 10.1016/j.compag.2013.05.004.

[3] "Feeding Nine Billion: The Issues Facing Global Agriculture." *THE ISSUES FACING GLOBAL AGRICULTURE* (n.d.): n. pag. *Croplife*. Web. 10 Mar. 2017. <https://croplife.org/wp-content/uploads/pdf_files/Feeding-Nine-Billion-The-Issues-Facing-Global-Agriculture.pdf>.

[4] Hung C., Nieto J., Taylor Z., Underwood J., Sukkarieh S. Orchard fruit segmentation using multi-spectral feature learning; Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); Tokyo, Japan. 3–7 November 2013; pp. 5314–5320.

[5] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In CVPR,
2005.

[6] Redmon, Joseph. "Darknet: Open Source Neural Networks in C." *Darknet* (2016): n. pag.

[7] Redmon J., Divvala S., Girshick R., and Farhadi A.: You Only Look Once: Unified, Real-Time Object Detection, CVPR 2016

[8] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A Library for Large Linear Classification, Journal of Machine Learning Research 9(2008), 1871-1874. Software available at <http://www.csie.ntu.edu.tw/~cjlin/liblinear>

[9] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster {R-CNN}: Towards Real-Time Object Detection with Region Proposal Networks. In Advances in Neural Information Processing Systems. In NIPS, 2015.