A Robust Mature Tomato Detection in Greenhouse Scenes Using Machine Learning and Color Analysis

Guoxu Liu
Dept. of Electronics Engineering
Pusan National University
Busan, Rep. of Korea
201693257lgx@pusan.ac.kr

Shuyi Mao
Dept. of Electronics Engineering
Pusan National University
Busan, Rep. of Korea
msy0725@pusan.ac.kr

Hui Jin
Dept. of Electronics Engineering
Pusan National University
Busan, Rep. of Korea
lay1007@pusan.ac.kr

Jae Ho Kim, Professor
Dept. of Electronics Engineering
Pusan National University
Busan, Rep. of Korea
jhkim@pusan.ac.kr

ABSTRACT
A new algorithm for automatic tomato detection in regular color images is proposed, which can reduce the influence of illumination, color similarity as well as suppress the effect of occlusion. The method uses a Support Vector Machine (SVM) with Histograms of Oriented Gradients (HOG) to detect the tomatoes, followed by a color analysis method for false positive removal. And the Non-Maximum Suppression Method (NMS) is employed to merge the detection results. Finally, a total of 144 images were used for the experiment. The results showed that the recall and precision of the classifier were 96.67% and 98.64% on the test set. Compared with other methods developed in recent years, the proposed algorithm shows substantial improvement for tomato detection.

CCS Concepts
• Computing methodologies → Vision for robotics • Computing methodologies → Object detection • Computing methodologies → Support vector machines

Keywords
Tomato detection; harvesting robots; machine learning; color analysis

1. INTRODUCTION
With the development of modern agriculture, intelligent agriculture has attracted more and more attention around the world. Among these, fruit harvesting robot is a rapidly developing branch due to its potential efficiency. For the harvesting robot, the first and a critic step is to detect the fruits autonomously. However, it is very difficult to develop a vision system as intelligent as human for the fruit detection. There are many reasons for this like uneven illumination, non-structural field, occlusion and some other unpredictable factors [1].

In recent decades, intensive efforts have been carried out on fruit detection research for harvesting robots. Bulanon et al. [2] proposed a color-based segmentation method for apple recognition. Luminance and red color difference in YCbCr model were used in their work. Another color space $L^*a^*b^*$ was employed to extract ripe tomatoes in [3]. These methods used only color features for fruit detection. So they relied heavily on the effectiveness of the color space used. However, it was difficult to select the best color model for color image segmentation in real cases [4]. Furthermore, relying only on color features causes losing much other visual information in the image which was proved very efficient for object recognition [5]. Kurtulmus et al. [6] proposed a green citrus detection method under natural outdoor conditions combining circular gabor texture features and eigenfruit, and 75.3% accuracy was reported. This method used several fixed thresholds for detection.

On the other hand, to overcome the problems of illumination variation and occlusion, some researchers have attempted to use various of sensors for fruit detection [7], [8]. Tanigaki et al. [7] used red and infrared laser scanning sensors to locate cherries on the tree, which can prevent the influence of the sunlight. Xiang et al. [8] employed a binocular stereo vision system for tomato recognition. Xiang argued that 87.9% of tomatoes were recognized correctly. These technologies usually provide better results than conventional RGB color image based methods. This is mainly due to the fact that similar reflectance in visible light frequency band may show a different result in non-visible band. Nevertheless, the high cost of the sensors makes it difficult to be commercialized.

With the development of machine learning algorithms, more and more researchers started to adopt machine learning in computer vision tasks including fruit detection [1]. Ji et al. [9] proposed a classification algorithm based on support vector machine for apple recognition. The recognition success rate was 89%. In [10], tomato fruits detection was implemented using image analysis and decision tree models, and 80% tomatoes were detected. Kurtulmus et al. [11] conducted a comparison experiment of peach detection in natural illumination with several classifiers including statistical classifiers, a neural network and a support vector machine classifier, combined with 3 image scanning methods. In [12], a method combining adaboost classifier and color analysis was developed for automatic tomato detection.

On the other hand, the Histograms of Oriented Gradients (HOG) was proposed for pedestrian detection [13]. The paper reported HOG feature was better than other features in detecting pedestrians.

ICMLC'11, February 22–24, 2019, Zhuhai, China.

DOI: http://dx.doi.org/10.1145/12345.67890

Therefore, the authors want to use HOG for improving fruit detection rate. The proposed system is as follows. For training, HOG and SVM are used for building the Detection Block (DB). For classification, a Basic Detection System (BDS) is proposed. It consists of DB, and False Color Removal (FCR). The input image is applied to one BDS, and the down scaled input images are supplied to several other BDSs. The outputs of all BDSs are processed with Non-Maximum Suppression (NMS) for getting the final results.

The remainder of this paper is organized as follows. Section 2 describes the proposed tomato detection methods. Section 3 reports and discusses the experimental results obtained using the proposed algorithm. Section 4 draws conclusions from this paper.

2. MATERIALS AND METHODS

2.1 Image Acquisition and Pre-processing

To develop and evaluate the proposed algorithm, images of tomatoes in the greenhouse were acquired in late December 2017 at the Vegetable Expo Park, Shouguang, China. A total of 144 images were acquired using a color digital camera (Sony DSC-W170) with a resolution of 3648×2056 pixels. The distance for photograph ranged from 500 – 1000 mm which accords with the best operation distance for the harvesting robot. As shown in Figure 1, the growing circumstances of tomatoes in this work varied a lot, including (a) separated tomatoes, (b) multiple overlapped tomatoes and (c) tomatoes occulted by leaves, stems or other non-tomato objects. To speed up the image processing, all the images were resized to 360×202 pixels using the bicubic interpolation algorithm.

(a) Separated tomatoes

(b) Multiple overlapped tomatoes

(c) Occlusion by leaves and stems

Figure 1. The images with different conditions: (a) separated tomatoes, (b) multiple overlapped tomatoes, and (c) occlusion by leaves and stems.

2.2 The Dataset

A total of 144 images were used for the experiment. In order to train the Support Vector Machine (SVM) classifier, 59 images were randomly selected from the captured images, and the rest 85 images were used for test. From the training images 137 tomato samples and 769 background samples were manually cropped to construct the training set. All the cropped samples were resized to 64×64 pixels to unify the size. The 137 tomato samples contained about 5 pixels of margin around on all the sides. The background samples were randomly cropped containing leaves, twigs, strings and other objects, and all the samples were labelled separately, 1 for the tomatoes and -1 for the backgrounds.

2.3 Detection Algorithm of Tomatoes

A flowchart of the developed tomato detection algorithm is shown in Figure 2, which can be summarized in the following several steps:

(1) Extracting the HOG features of the training samples
(2) For the HOG, SVM is used to build the Detection Block (DB)
(3) Sliding a small window on the input images for detecting the tomatoes.
(4) Extracting the HOG features of each window
(5) A Basic Detection System(BDS) consisting of DB and False Color Removal (FCR) is used and Tomato Candidates (TCs) are produced
(6) Several BDS are employed for the down scaled images
(7) TCs are merged by using Non-Maximum Suppression (NMS) method to get the final result

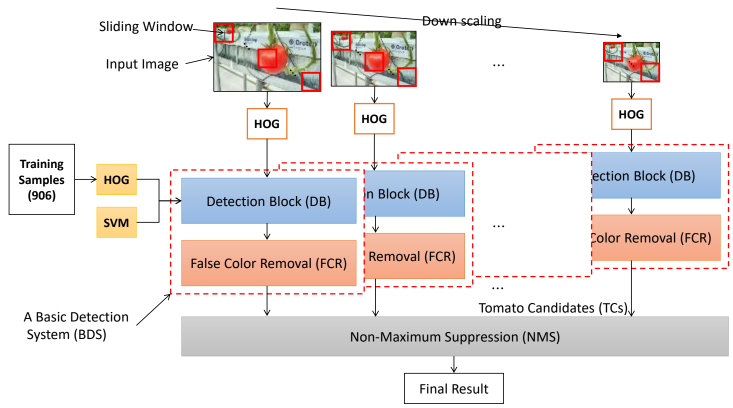


Figure 2. Flowchart of the proposed algorithm.

2.3.1 *Histograms of Oriented Gradients Feature Extraction*

Dalal and Triggs firstly proposed to use HOG [13] as a feature for pedestrian detection. Due to its efficiency in pedestrian detection, the HOG feature has been widely used. Firstly, the HOG can capture the shape information of an object and is invariant to geometric and photometric transformations. Secondly, the HOG can deal with the occlusion case. However, to the knowledge of the authors, there was few research on fruit detection using HOG. Thus, in this work, HOG features were used to evaluate its performance in tomato detection. HOG is a descriptor that encodes the shape of an object. It operates by dividing an image into a number of 8×8 pixel cells. For each cell a 1-D histogram of gradient directions or edge orientations over each pixel in the cell is calculated. All the histogram entries are combined to form the representation of the image. For better illumination invariance, a local response contrast-normalization method is employed, which is performed by accumulating a measure of local histogram energy over a 16×16 pixel block (4 cells) and normalizing all the cells of the block with the results. Figure 3 shows an example of HOG features for a tomato.

(a) An original image

(b) The gradient image

(c) The HOG descriptors visualization

Figure 3. An example of HOG descriptors: (a) An original image, (b) the magnitude of gradient image, and (c) the HOG descriptors visualization.

2.3.2 Theory of Support Vector Machine

Support Vector Machine (SVM) [14] is a strong classifier that uses a hypothesis space of all possible linear functions in a high dimensional feature space, trained with a learning strategy called margin maximization. It includes linear SVM and non-linear SVM.

2.3.2.1 Linear SVM

The principle of linear SVM is to find the hyperplane that can maximize the distance from the support vectors to the hyperplane. For instance, in Figure 4, the equation $\vec{w}\cdot\vec{x}+b=0$ denotes the separating hyperplane, and $f(x)=sign(\vec{w}\cdot\vec{x}+b)$ is the classifier decision function which equals +1 for positive samples and -1 for negative samples. The two positive samples (red) and one negative sample (blue) which lie on the margins are called support vectors. It is the support vectors that determine the separating hyperplane. In some cases, there are some outliers which cannot be separated linearly. In these cases, accepting a reasonable error, a slack variable ε_i is introduced to deal with the outlier data. For example, in Figure 4, it can be seen that two samples lie inside the margins, and the red one even goes over the separating hyperplane. These tow samples are treated as outliers. The decision function $f(x)$ is solved using Equation (1) – (3).

$$min_{\alpha}\ \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\,\alpha_j y_i y_j(\vec{x_i}\cdot\vec{x_j})-\sum_{i=1}^{N}\alpha_i \tag{1}$$

$$s.t.\quad \sum_{i=1}^{N}\alpha_i y_i=0 \tag{2}$$

$$0\le\alpha_i\le C,\quad i=1,2,\cdots,N \tag{3}$$

where α_i,α_j are the lagrange multipliers, $\vec{x_i}$ and y_i are the feature vector and label of sample i, respectively. C is the penalty parameter. There are N samples in all.

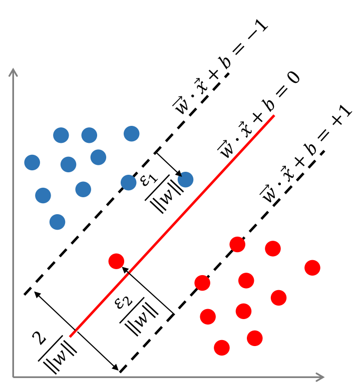


Figure 4. Linear SVM case.

2.3.2.2 Non-linear SVM

An appealing power of SVM is to cope with non-linearly separable questions. When the training data is non-linearly separable, SVM adopts a method called feature map to solve the problem. This method maps the original non-linearly separable feature space into a higher dimensional feature space which is linearly separable. Figure 5 shows a simple example. Feature map is carried out by using the kernel functions [15], which can perform the transform from the non-linearly separable space to a linearly separable space. The optimization problem to be solved is as the Equation (4) – (6).

$$min_{\alpha}\ \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\,\alpha_j y_i y_j K(\vec{x_i}\cdot\vec{x_j})-\sum_{i=1}^{N}\alpha_i \tag{4}$$

$$s.t.\quad \sum_{i=1}^{N}\alpha_i y_i=0 \tag{5}$$

$$0\le\alpha_i\le C,\quad i=1,2,\cdots,N \tag{6}$$

where α_i,α_j are the lagrange multipliers, $\vec{x_i}$ and y_i are the feature vector and label of sample i. And C, $K(\cdot)$ and N are the penalty parameter, the kernel function, and the total number of samples, respectively.

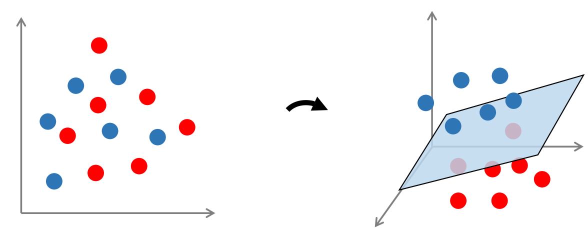
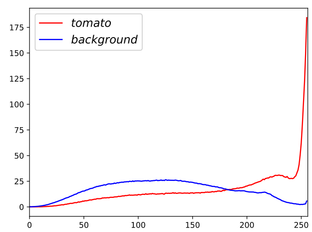


Figure 5. An example of the feature map. (Left: non-linearly separable in the original feature space, Right: linearly separable in a higher dimensional feature space)

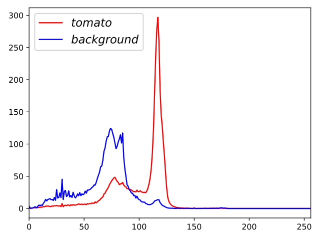
2.3.3 False Color Removal

Using the SVM classifier described previously, all sub-windows of the entire tomato image could be classified. However, there existed some false positive detections after the classification. Thus, a false positive elimination method is needed to reduce the false detections. Color features play an important role in fruit detection especially when the fruits have a different color from the background. In this work, the False Color Removal (FCR) is proposed for false detection elimination. The sub-window image was binarized using a color-based segmentation method. After the binarization, the ratio of white pixels among the whole window was calculated. If the ratio exceeded a threshold, this sub-window would be classified as a tomato. The threshold value used is 0.5 in this paper. Three color components from different color models – R, 1.5R-G (RGB) and H (HSI), were chosen for the experiment to distinguish tomatoes and backgrounds.

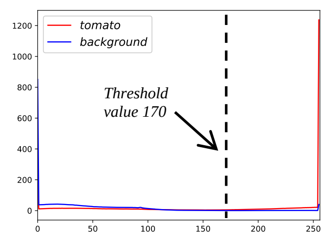
Totally, 1056 RGB samples from the dataset were used for the experiment. The histogram of each sample corresponding to each color component was calculated and then the average histogram for each component over all the samples was obtained. Finally, the color component which could best distinguishes tomatoes from backgrounds was chosen, along with the binary threshold. The results of the experiment were shown in Figure 6. It can be seen that all these three components can distinguish tomatoes and backgrounds. Compared with other two color components, the 1.5R-G component gave the best separation result. The final binarization threshold was found to be 170 for 1.5R-G through a trial and error method.



(a) Histogram of R component



(b) Histogram of H component



(c) Histogram of 1.5R-G component

Figure 6. The average histograms of three components over all the samples: (a) R component, (b) H component and (c) 1.5R-G component

3. RESULTS AND DISCUSSIONS

In this paper, several experiments were conducted to validate the performance of the developed method. The Detection Block (DB) was tested on a set of samples which consisted of 150 positive samples and 683 negative samples to evaluate its effectiveness. And the proposed method was compared with several other methods which were developed in recent years [6], [12]. Four indexes were used to validate the performance of the proposed algorithm. They are recall, precision, false positive rate (FPR), and F1 score defined by Equation (7) – (10), respectively. In this study, all experiments of the developed algorithm were performed on Python version 3.5 with an Intel® Core™ i5-4590 CPU@3.30 GHz.

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

$$FPR = \frac{FP}{Correctly\ identified + FP} \tag{9}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{10}$$

where TP, TN, FP, and FN are defined in Table 1.

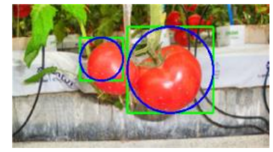
Table 1. Statistical analysis of the prediction and ground truth

		Predicted Class	
		Positive	***Negative***
True Class	***Positive***	*True Positive (TP)*	*False Negative (FN)*
	Negative	*False Positive (FP)*	*True Negative (TN)*

3.1 Results of SVM Classifier

With the HOG features, a sub-window based SVM classifier was developed for the task of tomato detection. The HOG features vary a lot for tomatoes and backgrounds. Combining the sliding window strategy and a SVM classifier, the features can be used to detect the tomatoes. A linear SVM was used in this work with the penalty parameter C=1. An example of before and after applying the SVM classifier was shown in Figure 7. From the figure, it can be seen that the two tomatoes were correctly detected with the inscribed circle (blue) of a bounding box (green).

(a) Original image

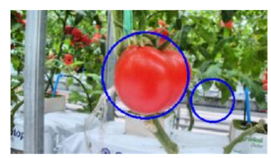


(b) Results of SVM classifier

Figure 7. An example of SVM classification: (a) before applying the SVM classifier and (b) after applying the SVM classifier.

3.2 Results of False Color Removal (FCR)

After detection using DB, the tomatoes can be found along with some false positives, i.e. the backgrounds. Thus the proposed FCR is then applied to reduce them. An example of before and after applying FCR was shown in Figure 8. It can be seen that the false positive in Figure 8(a) was successfully removed after color-based classification in Figure 8(b).



(a) Without FCR

(b) With FCR

Figure 8. Results of the FCR: (a) a result without the FCR and (b) a result with the FCR.

3.3 Accuracy of the Developed Detection Block (DB)

To evaluate the classification performance of the DB, the manually cropped tomato samples were used in the experiment. Both train and test sets were utilized. The results were shown in Table 2. It could be seen that the recall and precision on test set were 96.67% and 98.64%, respectively. This showed that the developed DB was competent for tomato detection.

Table 2. Accuracy of the Detection Block

Set	**Actual Categories**	**Samples Number**	**Classified Categories**		**Recall (%)**	**Precision (%)**
			Tomato	**Background**		
Train	Tomato	137	137	0	100	100
	Background	769	0	769		
Test	Tomato	150	145	5	96.67	98.64
	Background	683	2	681		

3.4 Comparison with Other Methods

To test the performance of the proposed method, two other methods which were recently proposed in papers [6], [12] were compared. An adaboost classifier which used haar-like feature as input and a method using circular Gabor Filter (CGF) and Eigen Fruit (EF) as features were compared with the proposed algorithm. Besides, another experiment in which all the steps were the same as the proposed except the color analysis step was set to test the effectiveness of color analysis. Table 3 lists the results of these methods. It can be seen that the proposed method achieved the second highest recall while it maintained the second highest precision. In addition, compared with the method using SVM classifier only, the precision improved a lot after color analysis. To provide a more objective assessment, F1 score was calculated which combined recall and precision together. Table 3 showed that the proposed method gave the highest F1 score compared with other methods. It demonstrated that the developed method is effective, and could be applied for mature tomatoes detection.

Table 3. Comparison of several tomato detection methods

Methods	Recall %	Precision %	Missed %	F1 %
SVM classifier only	**86.25**	57.74	13.75	69.17
Proposed	84.38	93.10	15.62	**88.52**
Adaboost [12]	80	71.51	20	75.52
CGF & EF [6]	78.75	**94.74**	21.25	86.01

4. CONCLUSIONS AND FUTURE WORKS

To overcome the difficulties harvesting robots faced in fruit detection, a novel algorithm is proposed in this paper. This method uses color images captured by a regular color camera. Compared with single feature detection methods, the proposed method used a combination of features including shape, texture and color information for fruit detection, which can reduce the influence of illumination, color similarity and occlusion factors. The HOG feature is adopted in this work. A SVM classifier is used to implement the recognition task combined with a sliding window and image pyramid based strategy, followed by a False Color Removal (FCR) method to eliminate false positives. At last, the widely used Non-Maximum Suppression (NMS) technology was employed to obtain the final results.

Several experiments were conducted to evaluate the efficiency of the proposed methods. 833 samples were used to validate the classification efficiency of the SVM classifier. The recall was 96.67% and the precision was 98.64%. It showed that the classifier with only HOG features can distinguish tomatoes from backgrounds very well. Comparing the proposed algorithm with other previous methods showed that the proposed method gave better results. This showed that the developed method was effective, and could be applied for mature tomato detection.

However, there are still some problems in the proposed method. The accuracy is not satisfactory for the overlapped and occulted tomatoes especially when the sheltered area exceeded 50%. Another limitation is that only one cultivar of tomatoes was experimented. Future research will focus on further improving the detection accuracy and extension to more other cultivars of tomatoes.

5. ACKNOWLEDGMENTS

This work was supported by BK21PLUS of Pusan National University. The authors would also want to give thanks to Vegetable Expo Park of Shouguang for the experimental images.

6. REFERENCES

[1] Zhao, Y., Gong, L., Huang, Y., & Liu, C. 2016. A review of key techniques of vision-based control for harvesting robot. *Computers and Electronics in Agriculture, 127* (2016), 311-323.

[2] Bulanon, D. M., Kataoka, T., Ota, Y., & Hiroma, T. 2002. AE—automation and emerging technologies: a segmentation algorithm for the automatic recognition of Fuji apples at harvest. *Biosystems Engineering*, 83, 4 (2002), 405-412.

[3] Yin, H., Chai, Y., Yang, S. X., & Mittal, G. S. 2009. Ripe tomato extraction for a harvesting robotic system. In *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on* (San Antonio, Texas, USA, October 11-14, 2009). 2984-2989.

[4] Wei, X., Jia, K., Lan, J., Li, Y., Zeng, Y., & Wang, C. 2014. Automatic method of fruit object extraction under complex agricultural background for vision system of fruit picking robot. *Optik-International Journal for Light and Electron Optics*, 125, 19 (2014), 5684-5689.

[5] Krig, Scott. *Computer vision metrics*. Springer, 2016.

[6] Kurtulmus, F., Lee, W. S., & Vardar, A. 2011. Green citrus detection using 'eigenfruit', color and circular Gabor texture features under natural outdoor conditions. *Computers and Electronics in Agriculture*, 78, 2 (2011), 140-149.

[7] Tanigaki, K., Fujiura, T., Akase, A., & Imagawa, J. 2008. Cherry-harvesting robot. *Computers and electronics in agriculture*, 63, 1 (2008), 65-72.

[8] Xiang, R., Jiang, H., & Ying, Y. 2014. Recognition of clustered tomatoes based on binocular stereo vision. *Computers and Electronics in Agriculture*, 106 (2014), 75-90.

[9] Ji, W., Zhao, D., Cheng, F., Xu, B., Zhang, Y., & Wang, J. 2012. Automatic recognition vision system guided for apple harvesting robot. *Computers & Electrical Engineering*, 38, 5 (2012), 1186-1195.

[10] Yamamoto, K., Guo, W., Yoshioka, Y., & Ninomiya, S. 2014. On plant detection of intact tomato fruits using image analysis and machine learning methods. *Sensors*, 14, 7 (2014), 12191-12206.

[11] Kurtulmus, F., Lee, W. S., & Vardar, A. 2014. Immature peach detection in colour images acquired in natural illumination conditions using statistical classifiers and neural network. *Precision agriculture*, 15, 1 (2014), 57-79.

[12] Zhao, Y., Gong, L., Zhou, B., Huang, Y., & Liu, C. 2016. Detecting tomatoes in greenhouse scenes by combining AdaBoost classifier and colour analysis. *biosystems engineering*, 148 (2016), 127-137.

[13] Dalal, N., & Triggs, B. 2005. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* (San Diego, CA, USA, June 20-26, 2005). Vol. 1, 886-893.

[14] Cortes, C., & Vapnik, V. 1995. Support-vector networks. *Machine learning*, 20, 3 (1995), 273-297.

[15] Boser, B.E., Guyon, I.M. and Vapnik, V.N., 1992. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory* (Pittsburgh, PA, USA, July 27-29, 1992). 144-152. ACM.