

# Enhancing Automated Defect Detection in Collagen Based Manufacturing by Employing a Smart Machine Vision Technique

Christopher D. Williams, Manoranjan Paul, and Tanmoy Debnath<sup>(✉)</sup>

Centre for Research in Complex Systems (CRiCS),  
Charles Sturt University, Bathurst, Australia  
{cwilliams,mpaul,tdebnath}@csu.edu.au

**Abstract.** Machine vision is now being extensively used for defect detection in the manufacturing process of collagen-based products such as sausage skins. At present the industry standard is to use a LabView software environment to manage and detect any defects in the collagen skins. Available data corroborates that this method allows for false positives to appear in the results which is responsible for reducing the overall system performance and resulting wastage of resources. Hence novel criteria were added to enhance the current techniques. The proposed improvements aim to achieve a higher accuracy and flexibility in detecting both true and false positives by utilizing a function that probes for the color deviation and fluctuation in the collagen skins. After implementation of the method in a well-known Australian company, investigational results demonstrate an average 26% increase in the ability to detect false positives with a corresponding substantial reduction in operating cost.

**Keywords:** Machine vision · Collagen · Defect detection · Robot vision

## 1 Introduction

Machine Vision is a technology used to replace the manual inspection of goods used in a wide area of manufacturing which has already become an increasingly important service. This technology is used in a variety of industries to automate production, increase production yield with greater efficiency, defect detection whilst improving product quality. Although the approach taken by each industry varies depending on the individual products, there are three main approaches for automatic visual inspection (AVI) and defect detection: modelling or referential, inspection or non-referential, and hybrid [7]. The inspection of printed circuit boards (PCBs) [3, 4], the defect detection of tiles [5] often use referential and hybrid approaches [1, 16]. On the other hand, in agriculture, the common approach for defect detection is by utilizing non-referential approaches [14]. This is mainly because problems arise in analyzing natural objects such as fruits because they have a wide variety of colours and textures [2]. Fruits belonging to the same commercial batch may have different colours depending on the stage of maturity.



**Fig. 1.** (a) Sausages, (b) collagen casing folds, and (c) an image of a flattened collagen casing (taken from [bbqhq.net](http://bbqhq.net), [diytrade.com](http://diytrade.com), and an Australian manufacturing company respectively)

Figure 1 shows sausages and their casings. Casings might have some defects such as creases or folds, black spec, etc. In an automated sausage manufacturing company, a collagen casing needs to be inflated and checked for defect before inserting materials into it. An Australian collagen casing manufacturing company uses a LabView AVI system [15] for accepting or rejecting casings to ensure quality by scanning the casings images. Major causes of the rejected images are due to the existence of creases or folds, black spots, and presence of hairs etc. However, manual inspections afterwards point out the limitations of the Lab-view program. It is noticed that a number of images are accepted although they should have been rejected and vice versa costing the company thousands of dollars in wasted resources. Our target is to reduce the false rejection rate thereby improving the overall defect detection. In order to achieve this goal an innovative non-referential image processing technique is proposed and implemented through real life experiments to improve the overall defect detection accuracy which is presented in this paper.

We utilize a modified method from PCB inspection [1] to include novel criteria for the defect detection of collagen sausage skins. The method works by locating abnormality in pixel intensity within a sausage image. Here a small abnormality is intensified so that by employing a predefined threshold our proposed method is able to detect it. This solution employs a local merit function to quantify the cost of a pixel that is defective [1]. The function comprises of two components: intensity deviation and intensity fluctuation. It is envisaged to increase accuracy by recognizing creases in the sausage skins created in the inspection process, as a non-defect as well as continuing to recognize black-specs as defects. Experimental results confirm that this approach of intensity deviation and intensity fluctuation improves the defect detection rates of collagen-based sausage skins significantly.

The remainder of this paper is structured as follows. Section 2 provides a brief description of how the defects are defined and inspection process of the collagen based products. Section 3 describes the proposed non-referential model for improving the detection process. In Sect. 4 experimental results and analysis are detailed. Finally Sect. 5 presents conclusions and directions for future work.

## 2 Collagen Skins

### 2.1 Definition and Features of Defects

During the manufacture of collagen-based sausage skins the most common defect is called black spec. These black specs are the remnants of hairs left over from the separation of the animal hide and the collagen film used for creating the sausage skins. All black specs larger than 1 mm are then picked up by the machine vision system and disregarded as defects in the system (Fig. 2).

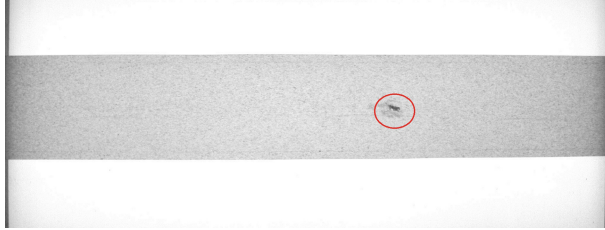


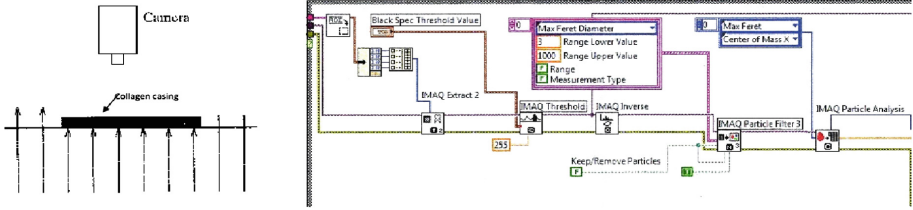
Fig. 2. Black spec defect larger than 1 mm in length.

### 2.2 The Inspection Process in the Existing LabView Defect Detection

Defect detection in the manufacture of collagen sausage skins requires light to be emitted through the bottom of the skin to shine through and illuminate any defects. This illumination technique is known as Transmission (Fig. 3). Here the rolled skins are inflated, flattened, and rolled over the top of a light source. The camera then inspects each region of interest (ROI) and relays the images to the computer system where currently LabView is employed to detect defect. The different steps of the method used by LabView to detect defects are as follows:

- Create a ROI: Using a set-algorithm that finds the edges (Edge Clamps) of the image, LabView determines the edge of the ROI.
- Black Spec Threshold value is determined to differentiate the background noise and the image.
- A LabView Particle Analysis Algorithm is used to determine the size and quantity of black specs.
- Depending on the user determined maximum size of the black spec, the image is then determined to be defective or not.

It can be seen in Fig. 3 that LabView allows the user to place each module in order to achieve the desired defect detection. It also demonstrates the lack of flexibility to manipulate the specific algorithms to customize details. The user can only drag and drop specific modules and add certain parameters for the operation. An issue created from this process occurs when the collagen skin



**Fig. 3.** (left) A transmission lighting system and (right) a schematic of LabView defect detection.



**Fig. 4.** A sample of sausage skin folding rejected by the LabView system, however manual inspection identified it as accepted, thus it a false positive case.

creasing as it is flattened and rolled through the vision inspection system. These folds are currently being wrongly picked up by the system as a defect and rejected (Fig. 4). This is a costly part of the manufacturing process and can be time consuming and expensive for manufacturers to overcome.

### 3 Proposed Non-referential Model

The proposed model is based on a *localized defects image model* (LDIM) [1], which uses a local merit function to quantify the cost that a pixel is defective. It utilizes two components that look at the intensity deviation and intensity fluctuation within the image. Taking into account of the fact that serious defects occur less often than slight defects, the paper states that further a defect’s intensity deviates from the dominant background intensity, the less and less often it is liable to occur. The complete model is expressed as follows:

$$C(x, y) = W_d \times D^2(x, y) + W_f \times F^2(x, y) \tag{1}$$

where the intensity deviation function is  $D(x, y) = V(x, y) - H_{x, y}$ , and the intensity fluctuation function is  $F(x, y) = \sigma(N(x, y)) - 4$ . Here -

- $C(x, y)$  is a merit function which represents the cost that a particular pixel is defective. It is the weighted summation of intensity deviation and fluctuation.
- $W_d$  and  $W_f$  are weights of  $D(x, y)$  and  $F(x, y)$  respectively. Both these weights are determined during the experiments which are  $\frac{1}{16}$  as suggested in [1].

- Dominant intensity of regional image  $H_{x,y}$  can be described by the most frequent intensity of the neighbouring pixels for the position at  $(x, y)$ . Thus intensity deviation  $D(x,y)$  can be expressed as the absolute difference between pixel value of source image  $V(x,y)$  and the dominant intensity  $H_{x,y}$ .
- Thus, intensity deviation  $D(x,y)$  at pixel  $(x,y)$  can be expressed as the absolute difference between pixel value of source image and dominant intensity.
- Evaluation of intensity fluctuation  $F(x,y)$  at  $(x,y)$  is based on the neighborhood pixels whereby the value of a pixel and its neighborhood can deviate from the main background intensity. Neighborhood pixels  $N(x,y)$  represents  $(2k+1)$  pixel at position  $(x,y)$ . Next standard deviation is applied to the  $N(x,y)$  to get the value of total fluctuation  $\sigma(N(x, y))$ .
- The difference between the intensity deviation, fluctuation and dominant intensity is the localized level:  $D(x,y)$  is at the pixel level,  $F(x,y)$  is at  $(2k+1)$  levels, while  $H_{x,y}$  is at regional level [1].  $k$  represents the pixel window size and its value is set as 5 pixels in our experiments as it was observed that this values provides optimum results.

### Pseudo code for the proposed model

```

Begin image input
Initiate parameters
  For all images
    For all pixel positions at  $(x,y)$ 
      Calculate  $D(x,y)$  and  $F(x,y)$ 
      Calculate model  $C(x,y)$ 
    End
  End
Calculate cost function above threshold
  Reject or accept image
End.

```

## 4 Experimental Results

The following results are garnered from three groups of test images sourced from a well-known Australian company. The company is a leading supplier of collagen casings for food, creating a range of sausage skins and other meat products. All images procured are from the production of sausage skins and are highly representable of the manufacturing facility. Each image is captured at  $2046 \times 800$  pixels at 72 dpi grayscale. Three groups of 50 images each were provided to us. After a manual inspection by the company of group 1s 50 test images, 34 and 16 were declared as defective and non-defective respectively. For group 2 the number of imperfect and perfect images was 30 and 20 respectively. The companys physical inspections have declared 42 and 8 images as faulty and non-faulty respectively in group 3. In order to validate the proposed non-referential model, at first a threshold value of 900 pixels was chosen as this represents

the number of pixels greater than the merit function threshold. Also included in our experiments where differing results are achieved through using different threshold pixel values (e.g. 500 and 2500 pixels). It is believed that images of different groups are captured in different times and for different setup of LabView machine and different initial materials, thus the thresholds are varied.

#### 4.1 Measurement of Results

To analyse the performance the results were split into the following three categories of classification: Specificity, Sensitivity and Accuracy. Specificity measures the proportion of negatives, which are correctly identified as such. Sensitivity calculates the proportion of actual positives, which are correctly identified as such. Accuracy is defined by the overall accuracy from the specificity and sensitivity.

$$\textit{Specificity} = (TN/(TN + FP)) \times 100 \% \quad (2)$$

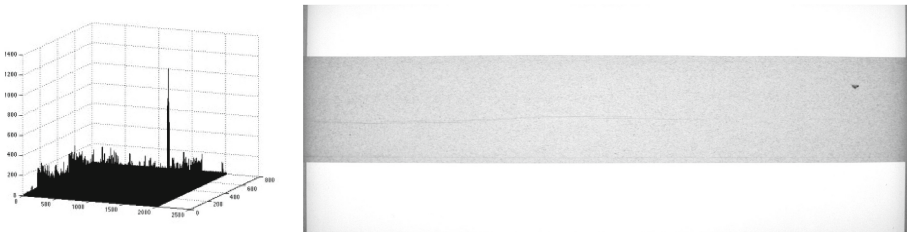
$$\textit{Sensitivity} = (TP/(TP + FN)) \times 100 \% \quad (3)$$

$$\textit{Accuracy} = ((TP + TN)/(TP + TN + FP + FN)) \times 100 \% \quad (4)$$

where, TN: True Negatives, TP: True Positives, FN: False Negatives, FP: False Positives.

Figures 5, 6, 8, and 10 in the next pages show graphs of the merit functions and the images that are representative for respective groups samples. Different cases are presented and compared with the LabView results to demonstrate our solutions supremacy. It can be seen from Table 1 that our algorithm has a significantly higher specificity. This demonstrates that it is able to correctly identify false positives. The overall accuracy reaches to 98 % for this batch of images.

Figure 5 shows that our algorithm works well to locate and identify the defect. The graph indicates a clear spike in the cost function, indicating a defect. Both LabView and our algorithm rejected this image correctly.



**Fig. 5.** An analysis of a defect image rejected by LabView, manual inspection, and also our solution. (left) Merit function values, (right) Actual image.

**Table 1.** Results of group 1. This table represents the LabView results, our solution, and the actual confirmed defects vs. false positives in the batch.

	LabView	Our results	Actual
Accept	3	14	-
Reject	47	36	-
TP	34	34	34
TN	3	16	16
FP	13	1	-
FN	0	0	-
Specificity %	18.75	94.12	-
Sensitivity %	100	100	-
Accuracy %	74	98	-



**Fig. 6.** Results of group 1. Showing a false positive defect. This image was found defective by LabView but shown correctly by our solution as non-defective. (left) Merit function values, (right) Actual image

Figure 6 shows a false positive case where the image has a typical creasing effect from the inspection process. The proposed algorithm worked as planned by indicating this image was a non-defect, whereas this image was rejected by the LabView environment.

#### 4.2 Threshold Values for Group 1 Images

Two thresholds (Th1 and Th2) were utilized to identify whether a particular image would be accepted or rejected. If the value of the merit function  $C(x,y)$  of a pixel position  $(x, y)$  is higher than the threshold Th1, that pixel is marked as defected. If the numbers of defect pixels are higher than the threshold Th2, the image is identified as defected. For an image  $I$ , the Th1 is set as follows based on the difference between the maximum and minimum pixel intensities and the *standard deviation* ( $\sigma$ ) of the image:

$$Th1 = \frac{1}{160}(\alpha^2 + \beta^2) \tag{5}$$

where  $\alpha = \max(I) - \min(I)$  and  $\beta = \alpha(I) - 4$ . Th1 was kept constant for all groups of images; however Th2 was varied based on the characteristics of the different groups images. The values in Fig. 7 show the different results obtained from changing the threshold (i.e., Th2) values in group 1. It is observed that by increasing the threshold to 2500 pixels the accuracy lowers to 83% from the default 900 pixels threshold, which scored 98%, whilst lowering our threshold from the standard 900 pixels to 500 pixels, the accuracy lowers to 87.2%.

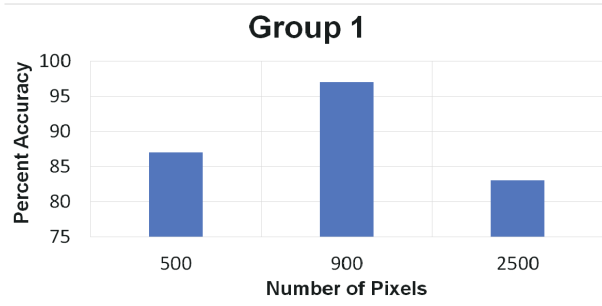


Fig. 7. Different accuracy levels at different threshold values for group 1 images.

Figure 8 presents an image which has small dark patches as well as a dark spec making the defect detection harder. The algorithm accepted this image as non-defect incorrectly. It is believed that the darkness of skin has allowed for a higher merit function value which is above the 900 pixels threshold. Thus, it is required to adjust our threshold for this kind of images so that they are accepted in our solution.

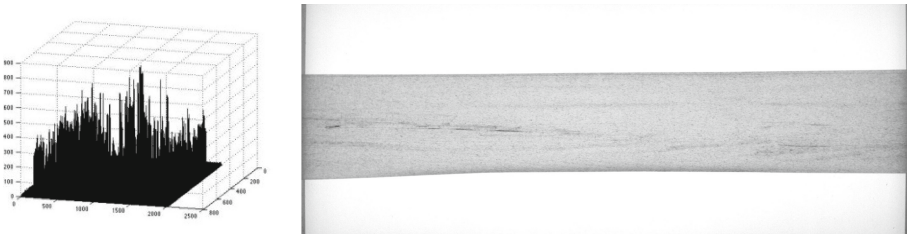


Fig. 8. Results of group 2 where it shows a defective image with black spec and creases that was incorrectly identified as a non-defective image. (left) Merit function values, (right) Actual image.

From Table 2 it is observed that the LabView solution rejected all images in this batch resulting in our algorithm having a significantly higher specificity and sensitivity, increasing our overall accuracy to 86% for this batch of images.

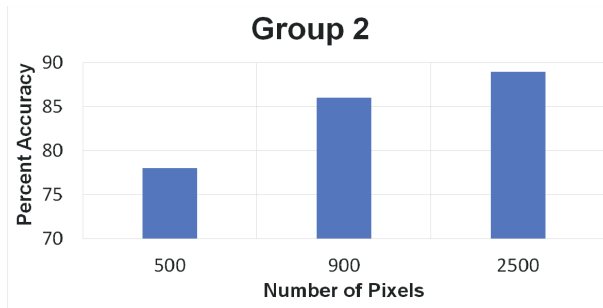


**Table 2.** Results of group 2. This table represents the LabView results, our solution, and the actual confirmed defects vs. false positives in the batch.

	LabView	Our Results	Actual
Accept	0	21	-
Reject	50	29	-
TP	30	26	30
TN	0	17	20
FP	20	3	-
FN	0	4	-
Specificity %	0	85	-
Sensitivity %	60	86	-
Accuracy %	60	86	-

### 4.3 Different Threshold Values for Group 2 and 3 Images

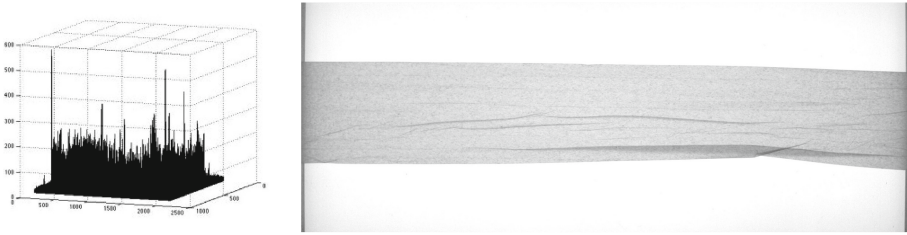
The values in Fig. 9 show the different results retrieved from changing the threshold values in group 2. It can be seen that by increasing the threshold to 2500 pixels the accuracy increases to 89.5% from the default 900 pixels threshold which scored 86%. Whilst lowering the threshold from the standard 900 pixels to 500 pixels, the accuracy decreases to 78%. It is believed that due to LabViews different parameter settings, a larger threshold is required for this group of images to improve the accuracy.



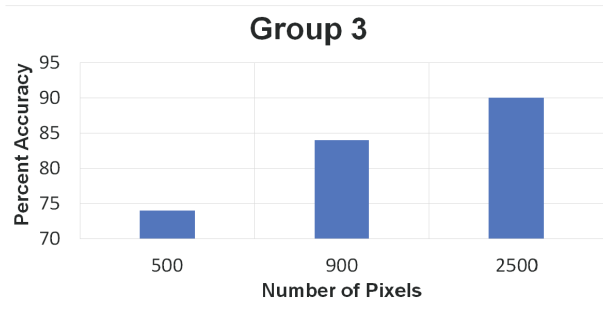
**Fig. 9.** Different accuracy levels at different threshold values for group 2 images.

Table 3 establishes that the LabView solution rejected all images in this batch whilst our algorithm recognized all 8 false positives and 8 true negatives. This resulted in an overall accuracy of 84%.

Figure 10 shows an image which is correctly accepted by the LabView environment but rejected by our solution, the most probable reason being that background intensity fluctuation was not high enough to breach our threshold level.



**Fig. 10.** The merit function and the corresponding image with creases which is a non-defective image however, it is rejected by our solution. (left) Merit function values, (right) Actual image.



**Fig. 11.** Different accuracy levels at different threshold values for group 3 images.

The values in Fig. 11 show the different results retrieved from changing the threshold values in Group 3. It is observed that by increasing the threshold to 2500 pixels the accuracy increases to 90 % from the default 900 pixels threshold which scored 84 %, whilst lowering the threshold from the standard 900 pixels to 500 pixels the accuracy decreases to 74 %.

In a nutshell, when the pixels threshold was lowered to 500 it yielded lower results consistently, as well as raising the threshold count to 2500 pixels had favourable results for two of the tests (groups 2 and 3). After comparing all the experimental results it can be surmised that although in some cases 2500 pixels threshold improves accuracy, in all cases a threshold of 900 pixels performs reasonably good.

## 5 Conclusions

The experimental results show that the current industry standard LabView environment tends to be less specific in its defect detection rates and would simply reject any product with a fold characteristic. This allows for a greater number of false positive results occurring during the inspection stage of manufacturing. On the contrary the proposed and implemented non-referential model achieves a significant increase in specificity, allowing for a higher proportion of negatives

to be correctly identified. The results demonstrate that whilst our solution doesn't fix the problem completely, it improves the overall accuracy of defect detection significantly. It is estimated that a general improvement of approximately 26% can save for the Australian Food processing company, an average of AUD 260 per day from the current AUD 1000 per day it costs for the wasted time and product involved in false defect detection in a single batch of processing. With factors such as moisture and humidity in the factory environment it is found that settings on the machine are manually adjusted to compensate. Thus, the algorithm threshold would need to be adjusting to suite. It is also noted that differences between results can vary depending on many factors such as:

- Darkness of the skin
- Crease length and amount of creases present in the skin
- Moisture in the collagen skin

Other characteristics for which our solution has detection difficulties include:

- Black spec could be located within the fold or creases of the skin. Depending on the density and amount of folds this defect has the ability to go undetected.
- Black spec could be located within the same frame as a finger defect.
- When a skin is severely creased throughout, our solution may not identify the intensity fluctuation or difference as high enough to trigger the defect threshold thereby rejecting the product.

In future it is envisaged to overcome the limitations of this work by improving the model so that all types of defects could be smartly detected and analysed.

**Acknowledgments.** We acknowledge the Australian Food processing industry to provide images in this analysis and experiments. Due to privacy and business issues we could not name the company in this manuscript.

## References

1. Xie, Y., Ye, Y., Zhang, J., Liu, L., Liu, L.: A physics-based defects model and inspection algorithm for automatic visual inspection. *J. Opt. Lasers Eng.* **52**, 218–223 (2014)
2. Lpez-Garca, F., Andreu-Garca, G., Blasco, J., Aleixos, N., Valiente, J.M.: Automatic detection of skin defects in citrus fruits using a multivariate image analysis approach. *J. Comput. Electron. Agric.* **71**, 189–197 (2010)
3. Mar, N.S.S., Yarlagadda, P.K.D.V., Fookes, C.: Design and development of automatic visual inspection system for PCB manufacturing. *J. Robot. Comput. Integr. Manuf.* **27**, 949–962 (2011)
4. Ibrahim, I., Ibrahim, Z., Khalil, K.: An improved defect classification algorithm for printing defects and its implementation on real printed circuit board images. *Int. J. Innovative Comput. Inf. Control.* **8**(5), 3239–3250 (2012)
5. Rahaman, A., Hossain, M.: Automatic defect detection and classification technique from image: a special case using ceramic tiles. *Int. J. Comput. Sci. Inf. Secur.* **1**(1) (2009)

6. Gruna, R., Beyerer, J.: Feature-specific illumination patterns for automated visual inspection. In: 12th IEEE International Instrumentation and Measurement Technology Conference, pp. 360–365, Graz, Austria (2012)
7. Moganti, M., Ercal, F., Dagli, C.H., Tsunekawa, S.: Automatic PCB inspection algorithms: a survey. *J. Comput. Vis. Image Underst.* **63**(2), 287–313 (1996)
8. Introduction to Machine Vision. <http://www.emva.org/cms/index.php?idcat=38>
9. Machine Vision. <http://www.machinevision.co.uk/what-is-it/4550546879>
10. Silva, H.G.D., Amaral, T.G., Dias, O.P.: Automatic optical inspection for detecting defective solders on printed circuit boards. In: 36th Annual Conference on IEEE Industrial Electronics Society, pp. 1087–1091, AZ, USA (2010)
11. Darwish, A.M., Jain, A.K.: A rule based approach for visual pattern inspection. *IEEE Trans. Pattern Anal. Mach. Intell.* **10**(1), 56–68 (2012)
12. Rokunuzzaman, M., Jayasuriya, H.P.W.: Development of a low cost machine vision system for sorting of tomatoes. *J. CIGR* **15**(1), 173–180 (2013)
13. Azriel, R.: Introduction to machine vision. *Control Syst. Magazines* **5**(3), 14–17 (1985)
14. Chen, Y., Chao, K., Kim, M.: Machine vision technology for agriculture applications. *J. Comput. Electron. Agric.* **36**, 173–191 (2002)
15. LabView Environment. <http://www.ni.com/labview/>
16. Aye, T., Khaing, A.S.: Automatic defect detection and classification on printed circuit board. *Int. J. Societal Appl. Comput. Sci.* **3**(3), 527–533 (2014)