An electronic component defect detection method based on SVM

Poxi Zhang¹, Weiqi Qian², Hongbin Cai¹*, Guanghui Lu¹

¹University of Electronic Science and Technology of China, Chengdu, China
²China Aerodynamic Research and Development Center, China

*Corresponding author: caihb@uestc.edu.cn

Abstract. In the production process of electronic components such as capacitors, bubble defects will occur inside the devices, which will lead to the loss of the functions of electronic components. Traditional methods for detecting bubble defects have many problems, such as low efficiency and low accuracy. Aiming at the bubble defects of the existing electronic components, an SVM-based detection method was proposed, which combined machine learning and image processing to complete the automatic detection of the bubble defects of electronic components. By using the feedback method, the error results obtained from the model classification are added to the training sample set, so that the problems of insufficient sample quantity and incorrect sample classification are solved. The accuracy of the proposed method was verified by capacitor component samples. The accuracy rate was 98.95%, the emission rate was 0.12%, and the error rate was 0.93%. It can be seen that the method proposed in this paper has a good effect on the defect detection of electronic components with the capacitor as an example.

Keywords: Defect detection; Image recognition; Support vector machines; Image process; Machine learning

1. Introduction

In the production process of electronic components, bubble defects of different sizes will occur inside the components. Such bubble defects will seriously affect the functions of the components. Therefore, the detection of bubble defects inside the components is of great significance in industrial production. The traditional bubble defect detection method is manual detection, which will lead to lots of problems. For example, it is the subjective judgment of the inspector to detect whether a component has defects, which may lead to wrong judgment. Due to the small size of electronic components themselves, manual detection has high requirements on human eyes and cannot work for a long time, which seriously affects the detection efficiency. To solve the problems existing in traditional detection methods, this paper proposes a machine learning detection method[1] based on SVM[2]. Since the bubble defect detection method proposed in this paper needs to be applied to electronic components with small volumes, the 0603 capacitors are taken as an example in the experiment (hereinafter referred to as the...
capacitor). The length of this capacitor is 1.60±0.15mm, and the width is 0.80±0.15mm. The experimental results obtained on this type of capacitor should apply to most small electronic components.

2. Image Process
Because of the small volume of capacitance, an ordinary camera is not able to ensure the clarity of the picture. To get high definition images, ultrasonic photography was used in the shooting, and the shooting results are shown in Fig. 1. As the figure shows, it is difficult to categorize all the capacitors in the large image directly, so the original image needs to be preprocessed[5]. Image preprocessing is mainly divided into two steps: the normalization part and the image segmentation part. The image preprocessing flow chart is shown in Fig. 2:

![Fig.1. Original Picture](image1)

2.1 Normalization Part
Since the input image may be obtained by different shooting equipment, the size of the original image may be slightly different, so the size of the original input image should be normalized first. Secondly, the original image is a color image, but the detection method in this paper does not need the color information, and the color image may introduce local lighting and other influences, so the image is greyed to avoid introducing unnecessary errors. Then, binarization is performed on the original image to obtain the binarization image. Because the binarization image has obvious corner points, the accuracy of edge detection based on the binarization image is much higher than that of the original image. Finally, a median filter is applied to the binarized image, which can effectively remove the noise points in the image and improve the accuracy of the classifier.

![Fig.2. Image Preprocessing Flow Chart](image2)

2.2 Image Segmentation
After normalized processing, noise such as ambient light and local light can be removed. Then, the image is segmented to obtain a single capacitor image that needs to be detected. First of all, edge
detection is carried out on the normalized output image. Because the edge of the binary image is obvious, the edge detection effect is very good. Based on the result of edge detection, the bounding rectangles of capacitors can be obtained, and both the coordinates and size of each rectangle are recorded. However, at this time, the bounding rectangles contain the bubble defect contour inside the capacitor. If the contour is directly used for segmentation, the bubble will also be segmented. Therefore, it is necessary to conduct bounding rectangle filtering to get the real bounding of capacitors. Through analysis of the actual situation of the capacitor bubble defect, it can be observed that the defect bubble inside the capacitor must be within another contour. Therefore, the inner contour can be removed if the contour is included in another contour. Besides, since the size difference of capacitors will not be too large, and the size normalization has been carried out in the normalized part, the area of a single capacitor contour obtained by segmentation should be similar in the image, so that the area can be used to filter the surrounding rectangle. In this way, all minimal bounding rectangles of capacitors can be obtained. After the final minimal bounding rectangles are obtained, the original image can be cropped by using the coordinates and size of each rectangle recorded, so that the original image can be divided into single capacitors, which are convenient for the next identification operation.

After a single capacitor is obtained, it needs to be processed. Because the original picture contains one thousand eight hundred capacitors, after the normalization part and image segmentation part, every single capacitor will have a slight difference, such as gray scale image characteristics. To reduce the error as far as possible, before the bubble defect detection, the size normalization and gray normalization processing of a single capacitor are carried out to ensure that all the gray scales of a single capacitor are within the same scale.

3. SVM-based Machine Learning Detection Method

To solve the problems of low recognition efficiency and subjective factors in artificial recognition, this paper proposes an SVM-based machine learning detection method. This method uses the theory of support vector machine to detect the defects of electronic components automatically and to ensure accuracy. The use of machines to replace manual operation effectively improves production efficiency.

3.1 Support Vector Machine Theory.

Support Vector Machine Theory is established based on Statistical Learning Theory[4](SLT). Statistical Learning Theory is used to study the rules of machine learning in small sample situations and has a solid theoretical basis. Support Vector Machine Theory is the main machine learning method of Statistical Learning Theory. Support Vector Machine Theory is widely used in text identification in complex background[5], handwritten font recognition[6], facial recognition[7], and other fields due to its ability to resist “high-dimensional disaster” and "over-learning".

Support vector machine can transform a problem to “solve quadratic programming with constrained condition” problem. For a classification problem, support vector machines look for a hyperplane in the sample space and divide the sample space into different sets. The hyperplane can be expressed as:

\[ \omega^T x + b = 0 \] (1)

Among equation (1), \( \omega = (\omega_1; \omega_2; \omega_3; \ldots; \omega_d) \) is a normal vector, which determines the direction of the hyperplane; \( b \) is the biases, which determines the offset to the origin. Therefore, the hyperplane can be determined by \( (\omega, b) \). Since the hyperplane has been determined, the distance from any point in the sample space to the hyperplane can be obtained as follows:

\[ d = \frac{|\omega^T x + b|}{||\omega||} \] (2)
Fig.3. Hyperplane Contrast

For a sample space, if the sample space is not large enough, there may be many hyperplanes that can correctly divide positive and negative samples. Support vector machine can select an optimal hyperplane from many hyperplanes that can correctly divide the sample space. In the opinion of the support vector machine, if a hyperplane can best split the sample space, the sum of the minimum distance between positive samples to hyperplane and negative samples to hyperplane should be as large as possible, to ensure the obvious distinction between positive and negative samples. As shown in Figure 3, it can be seen that the red hyperplane should be the hyperplane with the best effect.

Assume that hyperplane \((\omega, b)\) can correctly classify all sample \((x_i, y_i)\) in the sample space. Assume \(y_i \in (-1, 1)\), then:

\[
\begin{align*}
\omega^T x_i + b &\geq 1, y_i = 1 \\
\omega^T x_i + b &\leq 1, y_i = -1
\end{align*}
\]

(3)

According to (2) and (3), the sum of the minimum distance between the two sample spaces and the hyperplane is:

\[
\gamma = \frac{2}{\|\omega\|}
\]

(4)

\(\gamma\) is called margin. Support vector machines need to find the maximum interval under the constraint of (3). That is:

\[
\begin{align*}
\max_{\omega, b} &
\frac{2}{\|\omega\|} \\
\text{s.t. } &y_i (\omega^T x_i + b) \geq 1, \quad i = 1, 2, 3, ..., m.
\end{align*}
\]

(5)

To maximize the margin, \(\gamma\) should be as large as possible, that is, to minimize \(\omega\). Formula (5) can be rewritten as follows:

\[
\begin{align*}
\min_{\omega, b} \frac{1}{2} \|\omega\|^2 \\
\text{s.t. } &y_i (\omega^T x_i + b) \geq 1, \quad i = 1, 2, 3, ..., m.
\end{align*}
\]

(6)

Formula (6) is the basic formula of a support vector machine. Thus, the classification problem is transformed into a QP problem with constraint conditions. To solve the extreme value of (6), the Lagrange multiplier can be introduced, so that it is converted to solve its dual problem. The dual problem after the introduction of the Lagrange multiplier is shown as follows:

\[
L(\omega, b, \alpha) = \frac{1}{2} \|\omega\|^2 + \sum_{i=1}^{m} \alpha_i (1 - y_i (\omega^T x_i + b))
\]

(7)

Among (7), \(\alpha = (\alpha_1, \alpha_2, ..., \alpha_m)\). Take the derivative of formula 7 with respect to \(\omega\) and \(b\), and get the follows:

\[
\omega = \sum_{i=1}^{m} \alpha_i y_i x_i
\]

(8)
Substitute (8) into (7) and add the constraint of (9) to obtain the dual problem of (6):

\[
\max \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{j=1}^{m} \sum_{i=1}^{m} \alpha_i \alpha_j y_i y_j x_i^T x_j
\]

\[
\text{s.t.} \sum_{i=1}^{m} \alpha_i y_i = 0
\]

Formula (10) solves the linear separable problem. If the problem is nonlinear separable, the hyperplane needs to be projected onto the higher dimension. Therefore, (10) can be rewritten as:

\[
\max \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{j=1}^{m} \sum_{i=1}^{m} \alpha_i \alpha_j y_i y_j \phi(x_i)^T \phi(x_j)
\]

\[
\text{s.t.} \sum_{i=1}^{m} \alpha_i y_i = 0
\]

Introduce the kernel function \( \kappa(x_i, x_j) = \phi(x_i)^T \phi(x_j) \) to obtain the final support vector expansion:

\[
\max \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{j=1}^{m} \sum_{i=1}^{m} \alpha_i \alpha_j y_i y_j \kappa(x_i, x_j)
\]

\[
\text{s.t.} \sum_{i=1}^{m} \alpha_i y_i = 0
\]

Let \( \alpha' \) satisfy the condition obtained by solving (12), and substitute \( \alpha' \) into \( b = y_j - \sum_{i=1}^{m} y_i \alpha' \kappa(x_i, x_j) \).

Then, the final classification function is obtained:

\[
f(x) = \alpha'^T \phi(x) + b
\]

\[
= \sum_{i=1}^{m} \alpha' y_i \kappa(x, x_i) + b'
\]

Thus, using (13) to predict the input values \( x = (x_1; x_2; \ldots; x_n) \) and \( y = (y_1; y_2; \ldots; y_m) \), the classification results of support vector machine can be obtained.

3.2 Implementation of SVM-based Detection Method

The implementation steps of the SVM-based detection method in this paper are shown in Fig. 4.

Fig.4. SVM-based Detection Method Flow Chart

1) Image Preprocess

As shown in the image processing method in Chapter 2 above, the test samples and training samples should be pre-processed to avoid unnecessary errors and the original images should be converted into images suitable for the classifier.

2) HOG Feature Extraction

HOG feature extraction method[8] is used to extract the important feature items hidden in the image. In the SVM classifier, if all pixels in the sample are introduced, redundant parameters will be introduced which will lead to the high dimension of sample space. If only the important feature items extracted from the sample are used, the dimension of sample space can be effectively reduced, the
amount of calculation can be reduced, and the operation speed of the classifier can be improved. SVM is usually combined with HOG feature extraction. Such combination has many usages like human detection[9], vehicle logo recognition in traffic images, facial recognition.

HOG feature extraction principle mainly uses the pixel gradient in an image to distinguish foreground and background, which will extract features of interest. Specific method is as follows:

- Graying and gamma correction: Gamma correction will remove the influence of local overexposure and reduce the influence of local light intensity and local shadow on the subsequent step of the pixel gradient. In the process of obtaining the pixel gradient, the color information has little effect, so the image is grayed out to remove the color information.
- Image segmentation: When HOG feature extraction is carried out, we first need to select the area we are interested in and turn this part of the area into a window. HOG feature extraction will be carried out in the window. After selecting the appropriate window, divide the window into blocks of the same size, and then divide the block into smaller cells. The basic unit of HOG feature extraction is the cell. To calculate the pixel gradient, we slide each block and then accumulate the gradient histogram of each cell in each block. In the process of calculation, even though we have already undergone gamma correction in the first step has been underway for gamma correction, we still cannot rule out the influence of local lighting and shadow. So, we do not directly slide the whole block, but slide in a certain unit of length. Each calculation adds up the effect of the last local light or shadow. In this way, the error can be removed.
- Calculate the pixel gradient: For each pixel in the image, calculate the gradient value of each pixel, the specific calculation formula is as follows:

\[
\begin{align*}
G_x &= H(x+1,y) - H(x-1,y) \\
G_y &= H(x,y+1) - H(x,y-1)
\end{align*}
\]

(14)

The obtained pixel gradient and gradient amplitude are as follows:

\[
\begin{align*}
G(x,y) &= \sqrt{G_x^2 + G_y^2} \\
\alpha &= \arctan \frac{G_y}{G_x}
\end{align*}
\]

(15)

In (15), \(G(x,y)\) represents the amplitude and \(\alpha\) represents the direction of the gradient.

- Obtain the gradient distribution histogram of the final window: divide the gradient direction into \(N\) intervals (\(N\) is usually 9), and then add the calculated pixel gradient amplitude to the corresponding interval according to the gradient direction to obtain the gradient distribution histogram of the final window.

3) The SVM-based Classifier

It can be seen from SLT that the support vector machine transforms the classification problem into a QP problem. To simplify the complexity of the QP problem, a Lagrange multiplier is introduced to transform the QP problem into its dual problem. In practical problems, the classification problems encountered are usually linear inseparability. To solve the linear inseparability problems, two-dimensional classification lines can be converted into high-dimensional classification hyperplanes. To simplify the calculation of high-dimensional hyperplane, the kernel function \(\kappa(x_i, x_j) = \phi(x_i)^T \phi(x_j)\) is introduced. The kernel function can directly substitute the dot product in the high dimensional feature space, which greatly simplifies the computational complexity. So, the choice of kernel function is very important. Different kernel functions have different characteristics. The influence of different kernel functions on performance and classification effect is discussed in detail in [10]. In the experiment of this paper, the selected kernel function is the gaussian kernel function.

4) Feedback for Adjusting

The number of samples at the beginning of this experiment was limited. To provide the most accurate sample data for the SVM classification model as far as possible, a feedback adjustment part was added after each training, which added the wrong part of the classified results into the training
sample set. In this way, the wrong part of the samples obtained from each training can be corrected, to ensure that the model of the next training can be improved based on the model of the last training.

3.3 Experiment on SVM-based Classifier

The hardware platform tested in this paper is Intel(R) Core(TM) I7-7700K CPU, with a main frequency of 4.20GHz and a memory of 16G.

1) The Data Set

The data set used in this paper is derived from electronic component diagrams in the real production environment, and the capacitor is taken as an example for the test in this paper. The original capacitor image is shown in Fig. 1. To ensure the image quality, the image is obtained by ultrasonic photography technology. If there is a defect in the capacitor, there will be a cavity in the middle of the defective capacitor, and that is how we detect the defect. To analyze every single capacitor, the original capacitor graph is first segmented by using the image segmentation method to obtain a single target image to be detected. Then, every single target image was manually classified into two categories: defect samples and normal samples, as shown in Fig. 5.

![Sample Picture](image)

Fig.5. Sample Picture

Generating a large number of data sets may be difficult through manual classification only. Therefore, a part of the data set is first generated through manual classification, and then the sample is rotated and mirrored by the image processing method. In this way, the sample size can be greatly increased and can also ensure the accuracy of the sample data set. Through manual classification and image processing, the final number of samples is shown in the following table:

| Dataset      | Number of Normal | Number of Defect | Sum   |
|--------------|------------------|------------------|-------|
| Train        | 2792             | 3000             | 5792  |
| Dataset1     | 99               | 1645             | 1744  |
| Dataset2     | 33               | 1687             | 1720  |
| Dataset3     | 45               | 1737             | 1782  |

2) Experiment Result

In this paper, there are three main evaluation indicators for the test results: Accuracy (Acc), Missing rate (M) and Error rate (E). The evaluation methods of these three indicators are as follows:

\[
Acc = \frac{TN + TD}{TN + TD + EN + ED}
\]

(16)

\[
M = \frac{EN}{TN + TD + EN + ED}
\]

(17)

\[
W = \frac{ED}{TN + TD + EN + ED}
\]

(18)

Among (16), (17) and (18), TN represents the number of normal capacitors correctly detected. TD represents the number of defect capacitors wrongly detected as defect capacitors. EN represents the number of normal capacitors wrongly detected as normal capacitors. To test the accuracy of the model, the data sets different from the training samples were tested. Three test sets in the experiment were used to avoid the influence of the specificity of a test sample on the accuracy of the test results. The test results are shown in the following table:
Table 2 Experiment Result

| Dataset  | Acc  | M     | E     |
|----------|------|-------|-------|
| Dataset1 | 97.71% | 1.78% | 0.52% |
| Dataset2 | 98.95% | 0.12% | 0.93% |
| Dataset3 | 98.82% | 0.51% | 0.67% |

4. Conclusion
In this paper, the machine learning method of support vector machine is adopted to detect the defects in electronic components, and defect holes in electronic components are used as features to conduct model training. In order to extract local regional features of electronic components, HOG feature extraction is introduced, and finally, the classification model of support vector machine is obtained. The experimental results show that this classification model can correctly detect the defects in electronic components and obtain the correct classification results. Through the integrated support vector machine model detection and image processing methods, the final results can outline the contour of the defect areas in electronic components, which can easily be identified by users. In the experiment, there are also some problems. For example, in the process of segmentation of original images by using image processing technology, the content of segmentation may not be capacitor due to the introduction of noise points, which will lead to the classification process of introducing non-capacitor parts into the support vector machine. In the future, other machine learning methods are considered to directly recognize the original image. In this way, the image segmentation part is removed, and the non-electronic component image can be avoided as input to the classification model. In addition, as the number of training samples increases with the support vector machine model, the model parameters will also increase. So other methods can be considered to increase the classification speed of the model.

References
[1] Goodfellow I, Bengio Y, Courville A. Machine learning basics[J]. Deep learning, 2016, 1: 98-164.
[2] Suthaharan S. Support vector machine[M]. Machine learning models and algorithms for big data classification. Springer, Boston, MA, 2016: 207-235.
[3] Russ J.C. (1990) Image Processing. In: Computer-Assisted Microscopy. Springer, Boston, MA.
[4] V. N. Vapnik, "An overview of statistical learning theory," in IEEE Transactions on Neural Networks, vol. 10, no. 5, pp. 988-999, Sept. 1999, doi: 10.1109/72.788640.
[5] Chen D, Bourlard H, Thiran J P. Text identification in complex background using SVM[C]. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001. IEEE, 2001, 2: II-II.
[6] Rajput G G, Horakeri R, Chandrakant S. Printed and handwritten mixed Kannada numerals recognition using SVM[J]. International Journal on Computer Science and Engineering, 2010, 2(05): 1622-1626.
[7] Guo G, Li S Z, Chan K. Face recognition by support vector machines[C]. Proceedings fourth IEEE international conference on automatic face and gesture recognition (cat. no. PR00580). IEEE, 2000: 196-201.
[8] Mizuno K, Terachi Y, Takagi K, et al. Architectural study of HOG feature extraction processor for real-time object detection[C]. 2012 IEEE Workshop on Signal Processing Systems. IEEE, 2012: 197-202.
[9] Pang Y, Yuan Y, Li X, et al. Efficient HOG human detection[J]. Signal Processing, 2011, 91(4): 773-781.
[10] Patle A, Chouhan D S. SVM kernel functions for classification[C]. 2013 International Conference on Advances in Technology and Engineering (ICATE). IEEE, 2013: 1-9.