Volume 14 • Issue 2 • 2022
Engineering Management in Production and Services

sciendo

ISMSME International Society for Manufacturing, Service and Management Engineering

received: 12 December 2021
accepted: 5 May 2022

pages: 13-25

A novel data mining approach for defect detection in the printed circuit board manufacturing process

Blanka Bártová
Vladislav Bína

ABSTRACT
This research aims to propose an effective model for the detection of defective Printed Circuit Boards (PCBs) in the output stage of the Surface-Mount Technology (SMT) line. The emphasis is placed on increasing the classification accuracy, reducing the algorithm training time, and a further improvement of the final product quality. This approach combines a feature extraction technique, the Principal Component Analysis (PCA), and a classification algorithm, the Support Vector Machine (SVM), with previously applied Automated Optical Inspection (AOI). Different types of SVM algorithms (linear, kernels and weighted) were tuned to get the best accuracy of the resulting algorithm for separating good-quality and defective products. A novel automated defect detection approach for the PCB manufacturing process is proposed. The data from the real PCB manufacturing process were used for this experimental study. The resulting PCA-LWSVM model achieved 100 % accuracy in the PCB defect detection task. This article proposes a potentially unique model for accurate defect detection in the PCB industry. A combination of PCA and LWSVM methods with AOI technology is an original and effective solution. The proposed model can be used in various manufacturing companies as a postprocessing step for an SMT line with AOI, either for accurate defect detection or for preventing false calls.

KEY WORDS
quality management, defect detection, AOI, PCA, PCB, SVM

10.2478/emj-2022-0013

Blanka Bártová

Faculty of Management, University of Economics in Prague, Czech Republic
ORCID 0000-0003-4500-9941
Corresponding author:
e-mail: blanka.bartova@vse.cz

Vladislav Bína

Faculty of Management, University of Economics in Prague, Czech Republic
ORCID 0000-0002-9480-9489

Introduction

Quality inspection is a crucial stage in the assembling process of PCB manufacturing. It shows whether the board works correctly or not. Manual inspection of PCBs is laborious, time-consuming and imprecise as it is susceptible to human errors. Consequently, it is costly and ineffective. Currently, companies for PCB manufacturing use automated Surface-Mount Technology (SMT) lines to ensure better product quality and the manufacturing process continuity. The PCB

Bártová, B., & Bína, V. (2022). A novel data mining approach for defect detection in the printed circuit board manufacturing process. *Engineering Management in Production and Services, 14*(2), 13-25. doi: 10.2478/emj-2022-0013

manufacturing of the SMT assembly line goes through multiple steps of automatic handling. To ensure good quality and reduce the number of defects, advance inspection tasks, such as AOI, are becoming more popular. These quality inspection tasks are realised at different stages of the assembly process. The traditional defect detection methods have various disadvantages for application on big data sets, such as strong dependency on a designed template, time consumption, and high computational costs, which can be challenging for companies in the production environment (Hu & Wang, 2020). AOI placed on an SMT line inspects quality assurance of the processed PCB and, subsequently, can distinguish the chip assembly defects (Kim & Park, 2020). A digital camera and set of sensors are used in an AOI system for capturing the image and gathering data of each sample PCB product for further analysis. Due to the contactless measuring, the AOI tool is considered flexible, fast and effective compared to the usual electrical test equipment. Despite this, the AOI solution, in some cases, is not completely effective in defect detection and tends to report false positives. Some authors believe this to be caused by the natural limitation of AOI in the evaluation of visible defects only. Particularly, all observed visual differences are detected as defects, even though they can have no consequence on the actual functionality of a PCB (Soukup, 2010). Products evaluated by AOI as false-positive need to be manually recontroled using the human factor, which means additional costs. To make the detection of PCB defects more effective, a model should be proposed as an AOI postprocessing step to obtain better and more accurate results.

Therefore, the aim of this research is so to design a data mining model for effective recognition of defective and good-quality products. The AOI achieves an accuracy of 96.24 %. Therefore, the following research questions are posed: "Is it possible to use the SVM method and achieve a more effective solution for a quality recognition compared to AOI?", "Which SVM algorithm provides the best effectiveness?".

This paper is organised as follows. Section 1 introduces the current state of the solved problems and discusses various approaches used in PCB defect detection. Section 2 presents the used methodological approach, the source dataset, and evaluation metrics. Section 3 contains experimental results of the used individual algorithm settings. Section 4 compares the used models, discusses the key findings and defines the proposed model.

1. Literature review

Many authors have already examined the PCB quality control process, and most current papers focus on quality control using image processing. Most recently, Kumar, Shreekanth and Prajwal (2020) examined the effectiveness of different image processing algorithms in combination with the feature extraction method. Yin et al. (2019) proposed an improved local binary fitting level set method to improve the accuracy and efficiency of the PCB image segmentation. An automated defect detection approach for increasing the accuracy of the quality control process on PCB lines, which applies a SURF-based algorithm to AOI images, has been introduced by Hassanin et al. (2019). Chavan et al. (2016) proposed an innovative system based on image processing that combines various algorithms, such as Fault Detection Algorithm, Canny Edge Detection Algorithm and Contour Analysis. Wang, Zhao and Wen (2016) focused on detecting the PCB soldered dot using the image processing method. Kim and Park (2020) extracted two solder regions from a PCB image and then used a dual-stream CNN for defect classification. The proposed solution proves a higher performance and lower weight than can be obtained by conventional methods. The proposed method also improved the F1-score, reduced weight, and accelerated inference time compared to a single stream CNN. Hu and Wang (2020) introduced a deep learning PCB image detection approach, which builds a new network based on Faster RCNN. They also used the ResNet50 method together with Feature Pyramid Networks as the pillar for feature extraction, aiming for the effective detection of small defects on the PCB.

Zakaria et al. (2020) examined whether the machine learning approaches can significantly contribute to better PCB fault detection in the assembly line. They presented several different attitudes to PCB defect detection using various machine learning methods. This review showed that methods, such as random forests, neural networks, or probabilistic approaches, had been applied for PCB defect detection with the use of an AOI. But in the end, they concluded that the use of machine learning methods in PCB defect detection is rather minuscule. Reshadat and Kapteijns (2021) examined and compared different machine learning models applied to the output dataset from the AOI. They found that the K-Nearest Neighbors method achieved the best results for their case.

This research aims to propose an AOI process for better detection of low-quality PCBs. Defect detection is considered one of the essential requirements for quality control in PCB production. The independent AOI is inclined to often make false calls when the AOI evaluates the product as defective, but after a manual check, the product is reassessed as good quality. These false calls become expensive for the company when they are more frequent than correctly detected defective products. The AOI on the SMT line at the company that cooperated with this study realises almost 4 % of false calls, which is considered a high rate. The research authors aimed to find a solution or propose a model for the higher accuracy of defect detection. Based on the previous literature review (Bartova, Bina & Vachova, 2022), the chosen method for this classification task was the support vector machine (SVM). SVMs are currently a hot topic in the machine learning community, creating a similar enthusiasm now as previously encountered by Artificial Neural Networks. Far from being a panacea, SVMs yet represent a powerful technique with an intuitive model representation not only for outlier detection but for classification and regression in general (Meyer, 2020). In recent years, the SVM method has received considerable attention because of its superior performance in pattern recognition and regression (Cortes & Vapnik, 1995; Bores et al., 1995; Vapnik, 1995; Vapnik, 1997; Burges, 1998; Vapnik, 1999). The SVM method is useful for tasks such as defect detection and classification in manufacturing. Isa, Rajkumar, and Woo (2007) proposed a model which combines Discrete Wavelet Transform and Support Vector Machine for sensor data processing and further oil and gas pipeline defect classification. Ghosh et al. (2010) investigated the SVM performance of pattern classification of defects from images. They proposed an SVM-based multi-class model for defect pattern recognition and inspection of commonly occurring fabric defects. Most recently, Mahfuz et al. (2020) explored the SVM model for feature selection to increase accuracy and reduce the false-positive rate in defect detection.

Compared to other machine learning algorithms, SVM appears to be a suitable candidate for several reasons: high accuracy achieved in similar classification tasks, generalisation ability without source data limit preconditions, fast learning and evaluation, and, last but not least, its flexibility (Zhang et al., 2005, Zhang & Zhang, 2001). To improve the accuracy of the SVM method, some methods for data preprocessing can be used. To deal with the data complexity and diversity, Sun et al. (2013) used PCA and particle swarm optimisation (PSO) together with SVM within the analogue circuit fault diagnosis task. They applied PCA and data normalisation as preprocessing steps, then SVM for diagnosis itself, and PSO was finally used to optimise the penalty and the kernel parameters of SVM.

2. Research methods

The dataset used for this empirical study comes from the AOI system developed by Saki Corporation, whose four digital multifrequency projectors provide accurate 3D measurements for high-quality images. Based on these images, AOI evaluates the quality of the product and categorises it as either good-quality or defective. The defective products are then manually checked by a manufacturer and categorised as either defective or falsely categorised as such. The source dataset has 63093 products in total (0.22 % defects, 3.76 % false calls and 96.02 % quality products). Since the number of false calls is rather high in comparison with defective products, this study focused on the improvement of the quality evaluation process.

Based on a previously developed PRISMA-based systematic review (Bartova, Bina & Vachova, 2022), the method Support Vector Machine (SVM) was chosen for further research on effective defect detection. The PCA method for feature extraction was used as a preprocessing step. Based on Mujica et al. (2008), the methods for the dimensionality reduction of a data set are especially beneficial for working with high volume data.

This research is based on a combination of these two methods into one algorithm sequence. The research authors aimed to find the most effective type of the SVM algorithm and rate the effectiveness and accuracy of the proposed models.

2.1. Data set description

This study used a data file from the AOI line from an unnamed company, where the fitting of PCBs is automated on the SMT line. At the end of the assembling process, a control process was performed using the AOI technology. The data set had 63 396 products and 217 variables. The distribution of the products can be found in Table 1. For the model, the “false calls” products were reclassified as “pass” since they were good quality but misclassified by the AOI.

Tab. 1. Source dataset distribution

	PASS	FAIL	FALSE CALL
Products count	60582	165	2649
Percentage	2.56 %	2.26 %	2.18 %

2.2. Data preprocessing

Classification problems in quality assurance were characterised, for example, by many contributing features, considering the training set size or the imbalanced distribution of the dependent variable (Rokach & Maimon, 2006). The authors of this study faced analogical problems in their source dataset. For this reason, it was necessary to preprocess the data for better handling in the experimental phase of the research. The first step was to delete variables for which most of the data were missing. It was found that 148 variables did not contain values for more than 60 000 products; therefore, they were removed from the data file. Once constant and unimportant variables were removed from the remaining group of 68, only 25 variables were left. Then, the missing values were imputed by the predictive mean matching method. However, the dimensionality in the data set was still relatively high; consequently, the PCA analysis was used for feature extraction.

2.3. Model's evaluation

The most critical factor was the accuracy of the model since the research aimed to correct the classification of as many products as possible. The accuracy of different types of used SVM models was tested using the confusion matrix (Table 2).

Tab. 2. Confusion matrix

Confusion matrix		Actual class: Positive	Actual class: Negative
Prediction	Positive	TP (true positive)	FP (false positive)
	Negative	FN (false negative)	TN (true negative)

The accuracy measure commonly employed for classifier performance evaluation is defined by Eq. (1).

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

Nevertheless, with many present negative occurrences (in this case, good-quality products), it is useful to measure the performance by ignoring the correctly predicted negative items. In this case, well-known performance measures, such as precision (P), recall (R), or other factors, can be used (Rokach & Maimon, 2006). Several measures were used for the evaluation of models within this research.

Tab. 3. Accuracy measurements

Measure	Derivations	Index description
Recall	TPR=TP/(TP+FN)	How many items of the "not passed QA" class are correctly recognised
Precision	PPV=TP/(TP+FP)	How many items classified as "not passed QA" are true "not passed QA"
Specificity	SPC=TN/(FP+TN)	Expresses the proportion of products whose test is negative (quality products) among all those that actually have no defect
Negative Predictive Value	NPV=TN/(TN+FN)	The probability that following a negative test result, an individual product will truly have no defect
False Positive Rate	FPR=FP/(FP+TN)	The probability that a false call will occur and a positive result will be given when the true value is negative
False Discovery Rate	FDR=FP/(FP+TP)	The expected rate of Type I errors: the result is a false-positive
False Negative Rate	FNR=FN/(FN+TN)	The probability that a true-positive item will be missed by the test

The calculation formulas of these factors are presented in Table 3. The currently used AOI solution shows too many false calls, so the False Discovery Rate is high (4.18 %). This research aimed to create a model decreasing the value of this factor.

It also evaluated two more factors: the number of vectors needed for accurate model creation and the time for model counting. According to Tseng et al. (2015), a lower number of support vectors needed signifies the robustness of the classifier. This study assumed that the lower number of vectors was better. This also correlated with the duration of the algorithm execution.

2.4. Feature extraction phase

In machine learning tasks, each additional feature in the dataset exponentially increases the requirement of data points to train the model. The learning algorithm needs an enormous amount of data to search for the right model in the higher dimensional space. Therefore, this study used the PCA analysis for the reduction of variables in the data. This caused the data transformation into fewer dimensions and acted as the summaries of the features.

PCA reduces data by their geometrical projection into lower dimensions, and there arise the so-called principal components (PCs). The goal is to find the best summary of the data using a minimum number of uncorrelated PCs. The first PC minimises the total distance between the data and their projection onto the PC, in other words, the first PC explains the largest variability portion of the original data (Kakkar & Narag, 2007).

The eigenvalue variance was used to extract the number of PCs for this study. The analysis also provided the proportion of total variance in all variables accounted for each factor. It is evident from the data that the eigenvalues descended rapidly from the first value. The first component accounted for approximately 19 % of the variance of the original 25 factors, but subsequent components accounted for much less.

Thus, using the eigenvalue selection for this study, it can be assumed that only five factors were retained as PCs across all categories and questions. These PCs cumulatively accounted for approx. 65 % of the total variance. This is visible on the scree plot (Fig. 1).

It can be observed that the first five principal components can represent more than 60 % of the information stored in 25 used variables. The increment of the next variables is exceptionally low compared to them, so for further research, this study used only five variables (PC1-PC5).

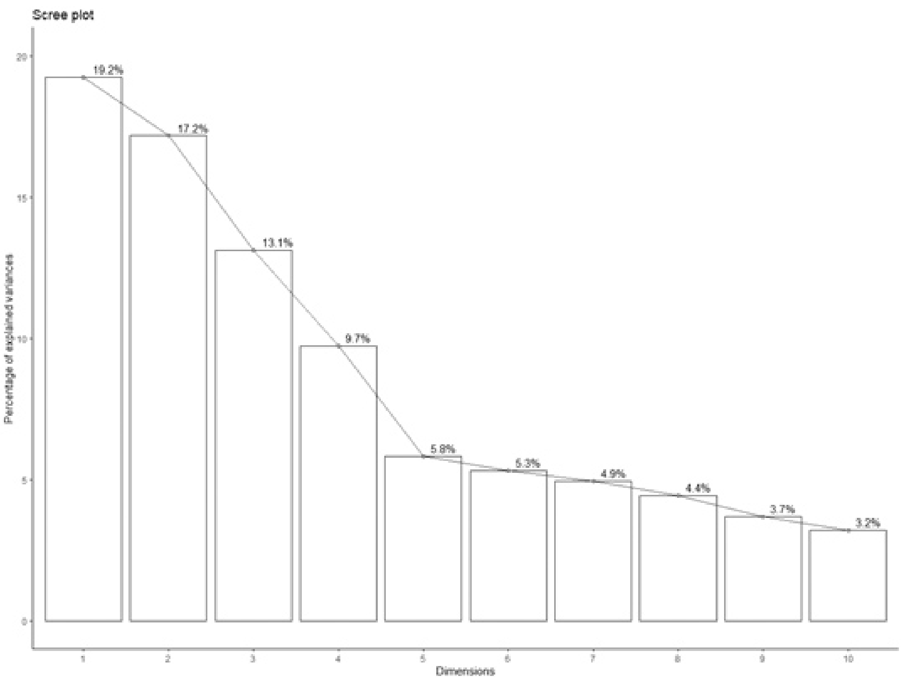

Fig. 1. PCA contribution — the scree plot

2.5. Support Vector Machine

Since SVM details are fully described in articles (Vapnik, 1998; Cristianini & Shawe-Taylor, 2000), this article offers a brief introduction to their fundamental principle. An SVM looks for the optimal hyperplane separating the two classes. The algorithm finds the optimal hyperplane by maximising the margin between the closest points of the two classes. For better work, mainly with non-linear data, kernel functions can be used. This research addresses the binary optimisation problem using a linear model, various kernels and weighted SVM models. Choosing different kernel functions produces various SVMs and may result in different performances (Burges, 1998; Aronszajn, 1950; Shawe-Taylor et al., 1998). Some work has already been done on limiting kernels using prior knowledge, but the best choice of a kernel for a given problem is still an open research issue (Williamson, Smola & Schölkopf, 1999; Chapelle & Schölkopf, 2002).

For SVM analysis, the data set was randomly divided into a training set (75 %) and a testing set (25 %). First, the model was tuned using the training data set; then, the created model was applied to the testing data set, and finally, the accuracy of the proposed model was evaluated. Different SVM algorithms were used for the prediction of defective products to achieve the best-fitted model.

3. Research results

At this stage of experiments, the research authors investigated the SVM models and their parameters for the successful detection of the defective products with the highest accuracy possible to find the most suitable model. Except for the linear SVM and different types of kernels, they also examined the weighted models.

For the use of the linear model and various kernels, different parameters were tuned, such as cost, gamma, and degree. The linear model had only one regularisation parameter C (cost). Parameter C controls the collation between variable misclassifications penalty and the margin width. A small value of the parameter C makes the constraints easy to ignore. This leads to a large margin. On the other hand, a large C value complicates the constraints disregard, which leads to a small margin. This parameter is also valid for all other models. For the purpose of finding the best model, the research authors tuned parameter C interval <0.01;100>. Unfortunately, the changing of the cost parameter did not influence the result accuracy at all.

When the data are not linearly separable, the various kernel functions can be used. The kernel functions are one of the important tricks of SVM. A kernel is a method of placing a two-dimensional plane so that it is curved in the higher-dimensional space (Boser, Guyon & Vapnik, 1992). There are several possibilities for the choice of this kernel function, including polynomial, sigmoid or radial basis (RBF). Additional parameter-slope gamma can be set for kernel models. Gamma is a hyperparameter that decides how much curvature we want in a decision boundary. When the parameter gamma is increased, then the decision boundary gets more curvature. First, the polynomial kernel was tried, which is a non-stationary kernel.

A kernel function represents the vectors' similarity in a feature space over polynomials of the original variables, allowing learning of non-linear models. In the case of the polynomial kernel, the value of cost and also the degree parameter of the SVC class need to be filled. However, the accuracy of the polynomial model does not reflect the changes in the used parameters. The best-achieved accuracy by the kernel polynomial function is 0.9979712 (99.8 %), which is the same as from the linear model. The next model, RBF (Gaussian) kernel, comes from a family of kernels where a distance measure is smoothed by an exponential function (Suo et al., 2008). RBF is the most used type of kernel function, mainly because it has a localised and finite response along the entire x-axis. Also, all the quality products have been correctly detected in this model, but all the defective products were wrongly assumed as good quality.

The last kernel function in this study was sigmoidal. As can be seen from the results in Table 4, the sigmoidal model results reflect the parameter changes the most, but on the other hand, the best accuracy is not higher than in the previous cases. Also, in the case of sigmoid SVM, the best-achieved result was 99.8 %, but no defective product was detected correctly.

All tuned models achieved the same result. All of the 15 735 quality products were correctly classified as good. However, the case was not as good with defective products. Out of 38 defects, all were misclassified as good-quality products (Table 5). The accuracy of the linear SVM was 0.9979712 (99.8 %). The model was not sufficient for fulfilling the set goal, even though the accuracy was high because no defec-

Tab. 4. Unweighted models accuracy

#	Cost	Lin.	Gamma	Rad. basis	Sig.	Dg.	Polyn.
1	0.001	0.998	0.0001	0.998	0.998	1	0.998
2	0.01	0.998	0.001	0.998	0.998	2	0.998
3	0.1	0.998	0.01	0.998	0.997	3	0.998
4	1	0.998	0.1	0.998	0.997	4	0.998
5	10	0.998	1	0.998	0.996	5	0.998
6	100	0.998	10	0.998	0.996	6	0.998

Tab. 5. Confusion matrix — unweighted models

Confusion matrix		Actual class	
		Positive	Negative
Prediction	Positive	0	0
	Negative	38	15735

tive product was correctly detected. For this reason, the data was assumed as not linearly separable.

Several models of different kernel functions were made, but none of them had sufficient accuracy. For this reason, the study continued searching for a model with satisfactory accuracy, especially a model able to detect defects even at the expense of a false-positive test of a small number of good-quality products. Based on some authors, weighted SVM (WSVM) could perform well in these classification tasks (Banjoko et al., 2019; Xanthopoulos & Razzaghi, 2014; Yang et al., 2007); therefore, it was used in this study as well.

3.1. Weighted SVM

The basic idea of the Weighted Support Vector Machine (WSVM) is assigning a different weight to each data point according to its relative importance in the class. Then, different data points have different contributions to the learning of the decision surface (Yang et al., 2007). Using a weighted linear SVM is better on such a data set than the simple linear SVM. Two separated regularisation parameters C1 and C2 are used instead of one. The weight of the penalty for misclassifying a good-quality product sample is represented by both parameters C1 and C2. The formula of the weighted support vector machine is expressed by Eq. (2).

$$Minimize_{w,b}\left(\frac{\| w \|^2}{2} + C_1 \sum_{i=1}^{n_1} \xi_i + C_2 \sum_{j=1}^{n_2} \xi_j\right) \tag{2}$$

where n_1 (respective n_2) is the number of quality products (respectively, defect products) in the training data. The parameters are then counted as can be seen in Eq. (3) and Eq. (4).

$$C_1 = \frac{1}{n_1} \tag{3}$$

and

$$C_2 = \frac{1}{n_2} \tag{4}$$

Of course, there are several approaches to setting the optimal weights.

The weights are only required for the algorithm training and are no longer used when the trained model is employed to predict the class label in the encoding process.

3.2. Weighted linear model

In the case of this study, when only several products with some defects are available in the dataset, a much higher weight must be attributed to them. Otherwise, the same result would probably be received as in previously run basic models, so that all good-quality products are correctly detected, but all defective products are misclassified. The attempt was made to heuristically try the SVM using different weights and different core functions. Then, the accuracy of the designed models was evaluated. First, the model was trained with a linear function. Table 6 provides the results of six runs of the SVM with mentioned weights of classes. The model was created with 100 % accuracy using a 0.0004 weight for good-quality products and a 0.1618 weight for defective products. This model generated 3771 support vectors, which is rather many, but despite this, the training time was less than ten seconds, which is exceptionally good.

Tab. 6. Linear weighted model accuracy

#	W(PASS)	W(FAIL)	Accuracy	# of vectors	Time (s)
1	0.0000159	0.007353	0.9284551	15437	30:59
2	0.0000794	0.036765	0.9409149	8330	16:94
3	0.0001588	0.073530	0.9657025	6284	14:76
4	0.0003177	0.147059	0.9978426	4601	11:12
5	0.0003336	0.154412	0.9999560	4504	10:85
6	0.0003495	0.161765	1.0000000	3771	9:84

The following table provides a confusion matrix of the resulting compiled model (Table 7). It demonstrates that no product was misclassified using this model.

Tab. 7. Confusion matrix — the linear weighted model

Confusion matrix		Actual class	
		Positive	Negative
Prediction	Positive	38	0
	Negative	0	15735

Even though an optimal model was already found, weights were tuned for models with other functions to investigate whether it was possible to achieve a 100 % correct classification of the product quality with other models.

3.3. Weighted polynomial model

Different weights were tried with various parameter degree settings for a model using a polynomial function. The best-created model generated only 108 support vectors, and also, the training time was very short. However, as Table 8 demonstrates, the accuracy of this model was not sufficient compared to the previously mentioned model.

Tab. 8. Polynomial weighted model accuracy

#	Dg.	W(PASS)	W(FAIL)	Accuracy	# of vectors	T(s)
1	3	0.000350	0.161765	0.99943	1354	4:49
2	5	0.000350	0.161765	0.99982	260	1:83
3	6	0.000350	0.161765	0.99991	99	0:92
4	7	0.000350	0.161765	0.99978	61	1:15
5	6	0.000318	0.147059	0.99994	108	0:96
6	6	0.000477	0.220588	0.99981	73	1:08
7	4	0.000477	0.220588	0.99987	491	2:12

Table 9 summarises the confusion matrix of the best performed weighted polynomial model. Only one product was misclassified using this model and was incorrectly marked as defective even though it was of good quality. This is the Type I error.

Tab. 9. Confusion matrix — the polynomial weighted model

Confusion matrix		Actual class	
		Positive	Negative
Prediction	Positive	37	0
	Negative	0	15735

3.4. Weighted Radial Basis Model

Results of the WSVM model with a radial basis function are summarised in Table 10. This model also proved to have better accuracy compared to the unweighted models. However, in this case, the tuned class weights and the cost parameter achieved accuracy that was still slightly worse compared to both previously performed weighted models. It should also be underlined that the radial basis model shows some cost values, not only very high calculation time but also a high number of vectors. Generally, the lower value of the cost parameter causes the higher execution time and number of support vectors, and in contrast, the higher cost value shows better performance in both mentioned factors and also higher accuracy.

Tab. 10. Radial basis weighted model accuracy

#	W(PASS)	W(FAIL)	C	Acc.	# of vec.	T(s)
1	0.000159	0.073530	0.01	0.99759	43609	406:13
2	0.000159	0.073530	0.1	0.98593	43609	421:92
3	0.003971	1.838235	10	0.99943	753	6:26
4	0.000477	0.220588	0.1	0.98593	42926	424:05
5	0.000874	0.404412	10	0.99537	2093	17:14
6	0.015884	7.352941	0.1	0.99937	5120	40:07
7	0.015884	7.352941	100	0.99981	173	1:82

Tab. 11. Confusion matrix — radial basis weighted model

Confusion matrix		Actual class	
		Positive	Negative
Prediction	Positive	35	0
	Negative	3	15735

In Table 11, we can see the confusion matrix of the best weighted radial basis model, which achieved a 0.9998098 accuracy. Three products were wrongly detected as defective, even though they were of good quality. Even in this case, it is a first-order error.

3.5. Weighted sigmoid model

The last tuned model uses the sigmoid function. Several support vectors were generated for each algorithm run. The classification accuracy in both the training and testing data sets was noted. According to Table 12, the model's accuracy was significantly worse than in previous models. The best-achieved model showed an accuracy of only 74.89 %. The calculation time was rather high compared to the polynomial and linear weighted models, and the same could be said for the number of support vectors. Also, the cost parameter did not influence the evaluation factors markedly as it was with the radial basis model.

The worst results were achieved by the weighted sigmoid model, where 3951 products were incorrectly marked as poor quality (Order II error). On the other hand, nine products were wrongly classified as good quality (Table 13).

Tab. 12. Sigmoid weighted model accuracy

#	W(PASS)	W(FAIL)	C	Acc.	# of vec.	T(s)
1	0.003971	1.838235	10	0.748177	11362	138:65
2	0.000080	0.036765	100	0.746846	11590	132:33
3	0.008816	4.080882	100	0.748938	11334	143:06
4	0.015884	7.352941	1	0.747036	11315	134:62
5	0.013501	6.250000	0.1	0.733215	12696	143:03
6	0.162810	75.36765	0.01	0.736385	12475	139:75

Tab. 13. Confusion matrix — sigmoid weighted model

Confusion matrix		Actual class	
		Positive	Negative
Prediction	Positive	35	0
	Negative	3	15735

4. Discussion of the results

After the analysis of the obtained models and testing of the accuracy levels achieved by using different kernels, the following conclusions were drawn. Fifty models were created and checked according to parameters of accuracy, number of vectors and execution time. To compare the models, the accuracy was tested using the confusion matrix and several metrics, visualised in Fig. 2.

Type I error was shown by basic unweighted models that used not only the linear function but also all kernels. Although the accuracy of these models was rather high (99.8 %), all of them misclassified all the defective products and assigned them wrongly to the good-quality class. Since the correct detection of defective products was the main goal, the study continued by including the class weights in the model. The weighted models performed significantly better, and their comparison according to the different indexes can be seen in Fig. 2. Amongst all the models, the weighted linear kernel achieved a perfect 100 % of recall rate, while other kernels always misclassified some of the defective products. All three models (weighted linear, weighted polynomial and weighted radial) achieved 100 % precision, and interestingly, the weighted sigmoid model showed extremely poor

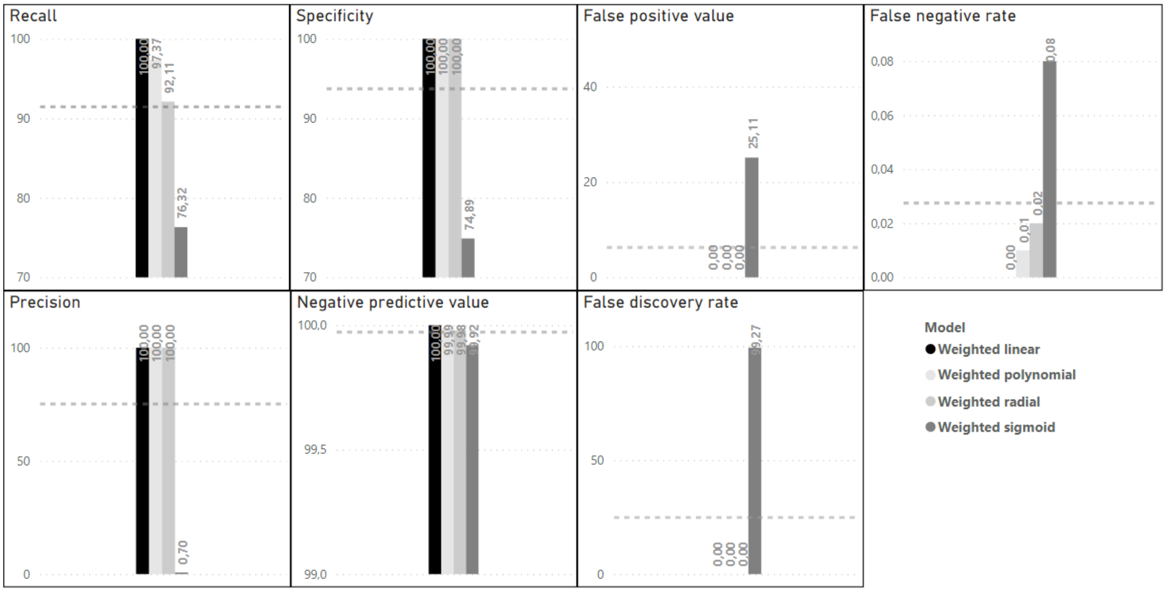

Fig. 2. Comparison of the weighted models' accuracy

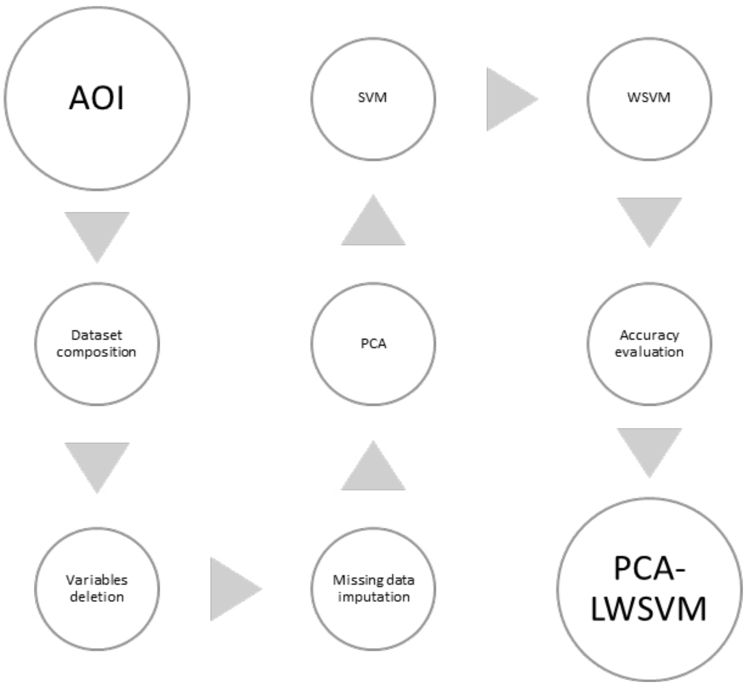

Fig. 3. Proposed model process

performance in this factor (<1 %). Differently from other used models, it had a remarkably high "false discovery rate", so this model mainly classified the products false positively (the weighted sigmoid model inclined to Type I error). In contrast, the weighted sigmoid model performed well compared to other models in the case of "negative predictive value". Based on the execution time performance and the number of supporting vectors, the best model was the polynomial weighted model. Unfortunately, this model showed worse accuracy than the linear weighted model. Consequently, the linear weighted model was evaluated as the best because the accuracy factor was definitely the most important for the study.

Fig. 3 demonstrates the used process leading to the resulting model. As was previously mentioned, the AOI on the SMT line for PCB mounting was used as the starting point from which the source data was gathered. The next step was the deletion of the unnecessary variables and missing data imputation for data preprocessing

Then, the PCA method was used for dimensionality reduction, and five final variables were created for further analysis. Based on the previous literature review, the SVM supervised algorithm was chosen for defect detection. Once it was found that the basic linear SVM and different kernels did not provide a satisfactory accuracy of classification, the addition of class weights was attempted. Weighted models performed much better. The Linear Weighted Support Vector Machine (LWSVM) model achieved 100 % accuracy. Therefore, the result of this study is the PCA-LWSVM model suitable for defect PCB detection implemented after image processing via AOI.

Conclusions

The paper presented different SVM algorithms that can be utilised for defective PCB detection on the output of the SMT line with AOI. This study aimed to investigate the optimal supervised parameters and feature representations. In the studied case, the weighted SVM model performed better than the linear SVM and different kernels. The resulting model combines the PCA feature extraction method and the WSVM classification algorithm. Different weights were tuned to find that 0.0003494449 for good quality products and 0.1617647059 for defective products proved to perform the best. AOI, which was originally used for defect detection, misclassified 4.18 % of samples and mismarked them as defective, while the proposed PCA-LWSVM model successfully classified both good-quality and defective products with a 100 % accuracy.

The main limit of the study can be the assumption that other models with a 100 % accuracy can be tuned and achieve even better performance from other points of view (the calculation time, weightless model, etc.). The weighted polynomial model performed very well and could be the subject of further investigation. Moreover, if certain data sets were used, the proposed model could be insensitive, and this means that different data sets may lead to various “suboptimal” models. It should also be mentioned that the proposed PCA-LWSVM model is hard to visualise. Another limitation is the range of training parameters C and Gamma. Higher values of these parameters can be used based on the data characteristics.

The obtained results can be further applied as a post-AOI procedure on the PCB automated assembly line. The proposed method helps manufacturers efficiently classify and manage defects in an automated optical inspection system in the surface-mount technology (SMT) line. The study is particularly useful for the automation of the quality control process since the manual retest of the wrongly classified products would be required no more.

Acknowledgements

The publication of the article for 11th International Conference on Engineering, Project, and Production Management - EPPM2021 was financed in the framework of the contract no. DNK/SN/465770/2020 by the Ministry of Science and Higher Education within the "Excellent Science" programme.

Ministry of Science and Higher Education
Republic of Poland

Literatura

Aronszajn, N. (1950). Theory of Reproducing Kernels. *Transactions of the American Mathematical Society, 68*, 337-404. doi: 10.2307/1990404

Banjoko, A. W., Yahya, W. B., Garba, M. K., & Abdulazeez, K. O. (2019). Weighted support vector machine algorithm for efficient classification and prediction of binary response data. *Journal of Physics: Conference Series*, 1366. doi: 10.1088/1742-6596/1366/1/012101

Bartova, B., Bina, V., & Vachova, L. (2022). A PRISMA-driven systematic review of data mining methods used for defects detection and classification in the manufacturing industry. *Production, 32.* doi: 10.1590/0103-6513.20210097

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, 144-152. doi: 10.1145/130385.130401

Burges, C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, *2*(2), 67-121. doi: 10.1023/A:1009715923555

Chapelle, O., & Schölkopf, B. (2002). Incorporating invariances in non-linear support vector machines. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems* (pp. 594–609). Cambridge, MA: MIT Press.

Chavan, R. R., Chavan, S. A., Dokhe, G. D., Wagh, M. B., & Vaidya, A. S. (2016). Quality Control of PCB using

Image Processing. *International Journal of Computer Applications*, *141*(5), 28-32.

Cortes, C., & Vapnik, V.N. (1995). Support-vector networks, *Machine Learning*, *20*, 273- 297. doi: 10.1007/BF00994018

Cristianini, N., & Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods.* NY Cambridge University Press. doi: 10.1017/CBO9780511801389

Ghosh, A., Guha, T., Bhar, R. B., & Das, S. (2010). Pattern classification of fabric defects using support vector machine. *International Journal of Clothing Science and Technology*, *23*(2/3), 142-151. doi: 10.1108/09556221111107333

Hassanin, A. A. I. M., Abd El-Samie, F. E., & El Banby, G. M. (2019). A real-time approach for automatic defect detection from PCBs based on SURF features and morphological operations. *Multimedia Tools and Applications*, *78*(24), 34437-34457. doi: 10.1007/s11042-019-08097-9

Hu, B., & Wang, J. (2020). Detection of PCB Surface Defects With Improved Faster-RCNN and Feature Pyramid Network. *IEEE Access*, *8*, 108335-108345. doi: 10.1109/ACCESS.2020.3001349

Isa, D., Rajkumar, R., & Woo, K. C. (2007). Pipeline Defect Detection Using Support Vector Machines. *6th WSEAS International Conference on Circuits, Systems, Electronics, Control and Signal Processing*, Egypt. Retrieved from http://www.wseas.us/e-library/conferences/2007egypt/papers/568-369.pdf

Kakkar, S., & Narag, A.S. (2007). Recommending a TQM model for Indian organisations. *The TQM Magazine*, *19*(6), 328-353. doi: 10.1108/09544780710756232

Kim, Y.-G., & Park, T.-H. (2020). SMT Assembly Inspection Using Dual-Stream Convolutional Networks and Two Solder Regions. *Applied Sciences*, *10*(13). doi: 10.3390/app10134598

Kumar, P., Shreekanth, T., & Prajwal, M. (2020). Automated Quality Inspection of PCB Assembly Using Image Processing. *International Journal of Image, Graphics and Signal Processing*, *12*(3). doi: 10.5815/ijigsp.2020.03.02

Mahfuz, R. A., M., Hoque, R., Pramanik, B. K., Hamid, E., & Ali Moni, M. (2020). *SVM Model for Feature Selection to Increase Accuracy and Reduce False Positive Rate in Falls Detection*. doi: 10.1109/IC-4ME247184.2019.9036529

Meyer, D. (2020). *Support Vector Machines*. Retrieved from https://cran.r-project.org/web/packages/e1071/vignettes/svmdoc.pdf

Mujica, L. E., Vehí, J., Ruiz, M., Verleysen, M., Staszewski, W., & Worden, K. (2008). Multivariate statistics process control for dimensionality reduction in structural assessment. *Mechanical Systems and Signal Processing*, *22*(1), 155-171. doi: 10.1016/j.ymssp.2007.05.001

Reshadat, V., & Kapteijns, R. A. J. W. (2021). Improving the Performance of Automated Optical Inspection (AOI) Using Machine Learning Classifiers. *2021 International Conference on Data and Software Engineering (ICoDSE)*. doi: 10.1109/ICoDSE53690.2021.9648445

Rokach, L., & Maimon, O. (2006). Data Mining for Improving the Quality of Manufacturing: A Feature Set Decomposition Approach. *Journal of Intelligent Manufacturing*, *17*(3), 285-299. doi: 10.1007/s10845-005-0005-x

Shawe-Taylor, J., Bartlett, P.L., Willianmson, R.C., & Anthony, M. (1998). Structural risk minimization over data-dependent hierarchies, *IEEE Trans. Information Theory*, *44*(5), 1926-1940. doi: 10.1109/18.705570

Soukup, R. (2010). *A methodology for optimization of false call rate in automated optical inspection post reflow*. doi: 10.1109/ISSE.2010.5547304

Sun, J., Wang, C., Sun, J., & Wang, L. (2013). Analog Circuit Soft Fault Diagnosis based on PCA and PSO-SVM. *Journal of Networks*, *8*(12), 2791-2796.

Suo, H., Li, M., Lu, P., & Yan, Y. (2008). Using SVM as Back-End Classifier for Language Identification. *EURASIP Jurnal Audio, Speech, and Music Processing*, 674859. doi: 10.1155/2008/674859

Tseng, T.-L., Aleti, K. R., Hu, Z., & Kwon, Y. (2015). E-quality control: A support vector machines approach. *Journal of Computational Design and Engineering*, *3*, 91-101. doi: 10.1016/j.jcde.2015.06.010

Vapnik, V. (1998). The Support Vector Method of Function Estimation. *Nonlinear Modelling*, 55-85. doi: 10.1007/978-1-4615-5703-6_3

Vapnik, V. N. (1995). *The nature of statistical learning theory*. New York, USA: Springer-Verlag.

Vapnik, V. N. (1999). An overview of statistical learning theory, *IEEE Trans. Neural Networks*, *10*(5), 988-999. doi: 10.1109/72.788640

Vapnik, V. N., Golowich, S., & Smola, A. (1997). *Support vector method for function approximation, regression estimation and signal processing. Advances in Neural Information processing Systems*. Cambridge, MA: MIT Press.

Wang, S. yuan, Zhao, Y., & Wen, L. (2016). PCB welding spot detection with image processing method based on automatic threshold image segmentation algorithm and mathematical morphology. *Circuit World*, *42*(3), 97-103. doi: 10.1108/CW-08-2015-0039

Williamson, R. C., Smola, A., & Schölkopf, B. (1999). *Entropy numbers, operators and support vector kernels*. Cambridge, MA: MIT Press.

Xanthopoulos, P., & Razzaghi, T. (2014). A weighted support vector machine method for control chart pattern recognition. *Computers and Industrial Engineering*, *70*, 134-149. doi: 10.1016/j.cie.2014.01.014

Yang, X., Song, Q., & Wang, Y. (2007). A weighted support vector machine for data classification. *International Journal of Pattern Recognition and Artificial Intelligence*, *21*(5), 961-976. doi: 10.1109/IJCNN.2005.1555965

Yin, Y., Luo, H., Sa, J., & Zhang, Q. (2019). Study and application of improved level set method with prior graph cut in PCB image segmentation. *Circuit World*, *45*(1), 55-64. doi: 10.1108/CW-03-2019-0028

Zakaria, S. S., Amir, A., Yaakob, N., & Nazemi, S. (2020). Automated Detection of Printed Circuit Boards (PCB) Defects by Using Machine Learning in Electronic Manufacturing: Current Approaches. *Materials Science and Engineering*, *767*. doi: 10.1088/1757-899X/767/1/012064

Zhang, C., Chen, X., Chen, M., Chen, S.-C., & Shyu, M.-L. (2005). A multiple instance learning approach for content-based image retrieval using one-class support vector machine. *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '05)*, 1142-1145. doi: 10.1109/ICME.2005.1521628

Zhang, L., Lin, F., & Zhang, B. (2001). Support vector machine learning for image retrieval, Proceedings of the IEEE International Conference on Image Processing (ICIP '01), 2, 721-724. doi: 10.1109/ICIP.2001.958595