Journal of Automation and Control, 2015, Vol. 3, No. 3, 96-100
Available online at http://pubs.sciepub.com/autoamtion/3/3/11

DOI:10.12691/automation-3-3-11

Diagnostics of Product Defects by Clustering and Machine Learning Classification Algorithm

Kamil Židek[1,*], Vladislav Maxim[2]

[1]Faculty of Manufacturing Technologies with a seat in Presov, Presov, Slovak Republic
[2]Institute of Automation, Robotics and Mechatronics, FME, Technical University of Kosice, Slovak Republic
*Corresponding author: kamil.zidek@tuke.sk

Abstract The article deals with usability of clustering and machine learning classification algorithm for search systematic surface errors. The main idea is to propose a methodology for the automated identification, diagnostics and localization of systematic errors in mass production. The introduced methodology consists of three levels: image processing for error parameterization, clustering for creating of errors classes (teach data) and prediction of new samples by machine learning algorithm. We conducted experiments with density based clustering algorithm DBSCAN. For classification we use multilayer perceptron MLP/ANN.

Keywords: *inspection, clustering, machine learning, image processing*

Cite This Article: Kamil Židek, and Vladislav Maxim, "Diagnostics of Product Defects by Clustering and Machine Learning Classification Algorithm." *Journal of Automation and Control*, vol. 3, no. 3 (2015): 96-100. doi: 10.12691/automation-3-3-11.

1. Introduction

Vision systems can be used in automated production processes for inspection, guidance, identification, measurement, tracking and counting, in many diverse industries. Vision systems may effectively replace human inspection in demanding cases such as nuclear industry, chemical industry, etc. In most cases, industrial automation systems are designed to inspect only known objects at fixed positions, characterized defects of faulty items and take actions for reporting and correcting these faults and replacing or removing defective parts from the production line [1].

At present, the development of computational performance embedded systems allows parallel image processing with advanced search for systematic errors by using clustering and machine learning classification algorithms. This data can then be used to identify the causes of these errors as well as continuous monitoring of product quality. This article describes automated methodology suitable for systematic error identification in mass production. We conducted experiment with clustering and classification algorithm. The first section describes advantages of selected clustering and classification algorithm. Main sections deal with description of methodology for automated systematic error diagnostics.

Implementation section contains brief description of prototype hardware and software solution.

2. Clustering and Classification Algorithm

In machine vision where it is supposed mass production we cannot use repeatedly clustering algorithm continuously for each new product error due to the large amount of data and the limitations of computing hardware.The solution is looking for algorithms combination from clustering and machine learning with fast prediction during mass production process. Clustering in our case tries to group a set of errors and find whether there is some relationship between its parameters. In general, the classification use acquired set of group classes for teaching and creates classification model and then can predict very fast to which class a new error belongs to. Clustering may then be repetitively performed with longer delays.

Algorithms can be divided to many groups for example hierarchical FLANN [2], density based DBSCAN and statistical K-means [3]. The most suitable is DBSCAN clustering algorithm because is it not necessary define number of clustering group. It is supposed that clustering algorithms provide finer distribution to classes like manually fixed supervised conditional setup of parameters [4,5].

Density-Based Spatial Clustering of Applications with Noise DBSCAN can find non-linearly separable clusters. Unlike other clustering algorithms that require many parameters, such as the number of clusters in the set, to be known and defined before computation, the DBSCAN algorithm has only two input parameters: the minimum size of a cluster and the maximum distance between points in a cluster. The algorithm operates by cycling through all points in the data set and calculating the number of neighbors each point has, which is defined as the number of other points that are within the minimum distance of the original point. Any data point that has fewer neighbors than the minimum cluster size parameter is declared to be a noise point that is not associated with any cluster. [6].

For example this dataset Figure 1 cannot be adequately clustered with k-means or Gaussian Mixture EM clustering.

Generalized DBSCAN is GDBSCAN and OPTICS, the fastest is hierarchical modification HDBSCAN [5,6].

Classification algorithm can be divided too to many groups of principles for example: statistical method, artificial intelligence, decision trees, boosted algorithms … etc [7-15]. In the next section are described chosen method based on ANN/MLP.

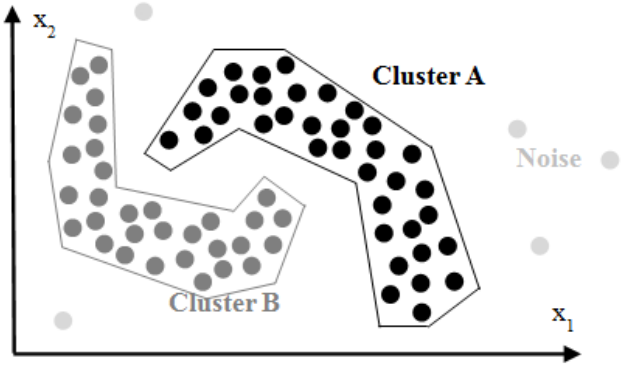

Figure 1. Dataset suitable for DBSCAN algorithm

Neural Network ANN/MLP. To classify the data into classes can also be advantageously used artificial neural network. Feedforward artificial neural network, or more specifically, a multi-layer Perceptron (MLP), the most commonly used type of neural network. MLP form consists of an input layer, an output layer, and one or more hidden layers. MLP each layer comprises one or more neurons associated with directionally neurons of the previous layer and also the next ones. The Figure 2 shows a three-layer Perceptron with three inputs, two outputs and hidden layer with three neurons [16].

All neurons in the MLP are similar and each has the stored value in the form of scales. The results of the neurons are transformed with the activation function f, which is usually the same but may be different for different neurons. The most commonly used activation function is symmetric sigmoid. As the previous algorithms, it is necessary learning using samples which must be placed at the entry to the exit pertaining to the output layer.

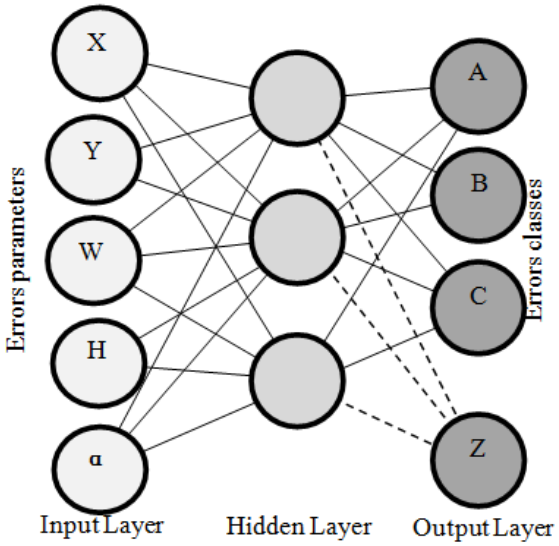

Figure 2. Example of multi-layer perceptron for classification

The main disadvantage is learning process of artificial neural networks which is extensively time consuming, especially with compare to each of presented machine learning classification algorithms.

3. Methodology of Error Clustering, Classification

Stopping production may be made primarily on the basis of a large number of errors in a row or on a time interval which we can show the error rate of production. This method is now used when stopped production on the basis of bad and good products, but the problem is that it tells us nothing about the character of the errors if it is a random or systematic error. If we cannot determine the character of the error is difficult to detect and remove the cause of their formation.

The clustering of large dataset is time consuming for desktop PC. It is not possible and necessary to run clustering for any new error detection in recognition loop. The solution can be combination of clustering method with classification teaching and prediction. The clustering provides slow basic assignment of similar errors to classes. Classification creates prediction model from training dataset provided by clustering. Trained classifier provides very fast assignment of new error to adequate class by prediction.

The main principle of surface error similarity diagnostics during operation we can divide to three steps:

- Image processing with contour detection – extraction of surface error with parameters,
- Search for similar errors by clustering algorithms, preparing of teach data,
- Create of classification model based on teaching and prediction of class assignment for new error.

Image processing algorithm for parameterization of detected errors is out of scope of this article and is in detail described in earlier article about error diagnostics [17].

We assume with these parameters of errors stored partially depended on three layers Figure 3.

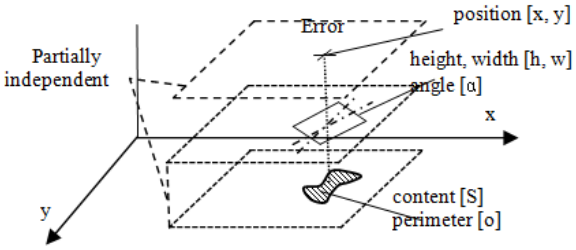

Figure 3. Extraction of parameters to partially dependent layers

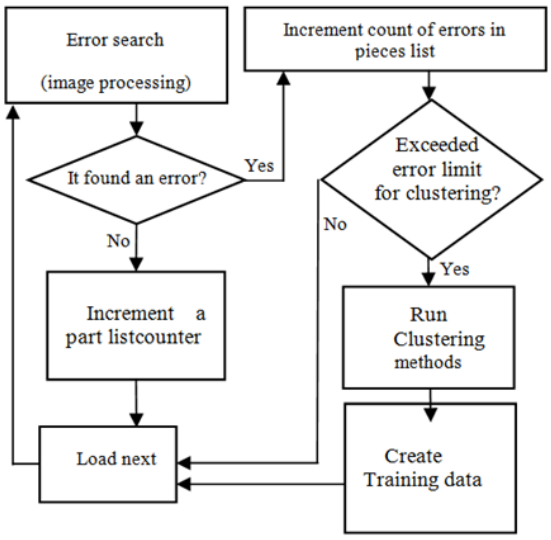

Figure 4. Principle algorithm of errors recognition

Second level of clustering algorithm threshold principle is shown on the Figure 4.

Principle for classification is universal for any proposed algorithm and is shown on the Figure 5.

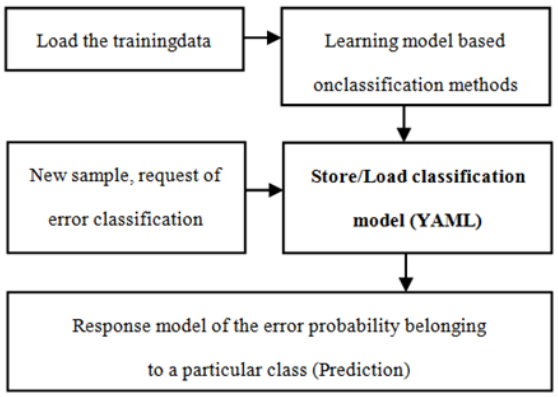

Figure 5. Classification of errors by teaching and prediction

Uses this methodology can help detect systematic errors from its location, size, perimeter, within components. The time variation of these parameters can predict degradation speed of errors (size increase or decrease).

4. Hardware and Software Solution of Vision System

The algorithms for clustering and classification were tested on two different platforms desktop PC (x86) and embedded (ARM) core: Intel Xeon x64 platform with (Ubuntu OS), and embedded board (SBC) with ARM architecture OlinuxinoA13(Fedora) and Raspberry PI (Raspberrian). Prototype stand with two different embedded systems and different camera sensors Firewire (1), USB camera (2) and CSI camera (3) is shown on the Figure 6.

Desktop PC uses Firewire camera, Olinuxino the USB web camera and Raspberry Pi CSI. The real-time vision task requires the industrial camera systems with Firewire for PC and high speed CSI interface for embedded systems.

For the vision recognition task exists many completed free or commercial libraries. For our task was selected OpenCV library (C++ version), because is free, open source, very fast, platform independent and includes current cutting-edge algorithms from image processing and machine learning (clustering, classification, prediction … etc.). The next reason was native support for algorithm acceleration with CUDA, OpenCL and multithreading TBB.

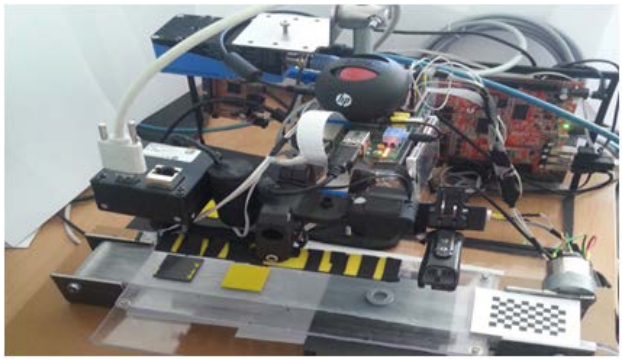

Figure 6. Experimental vision and image processing stand

Software solution of the vision system consists from two parts:

- Image processing and machine learning based on fast C++ Opencv library,
- Graphical user interface based on web technologies with AJAX support.

GUI uses PHP scripts as bridge between C++ and HTML5/jQuery web visualized pages. Web applications are platform independent that was a basic point to use these technologies. The next advantages are remote access and saving power of processing device because the performance is used by browser on client side.

Graphical web GUI interface for control diagnostics process, conditional and advanced clustering is shown on the Figure 7. We assume that error data from recognition process can be available simultaneous. This necessitates a network database to store data. Recognition results are stored in integrated SQlite3 database and can be accessed remote by web GUI. This set of error data in certain defined interval creates basic training set for clustering.

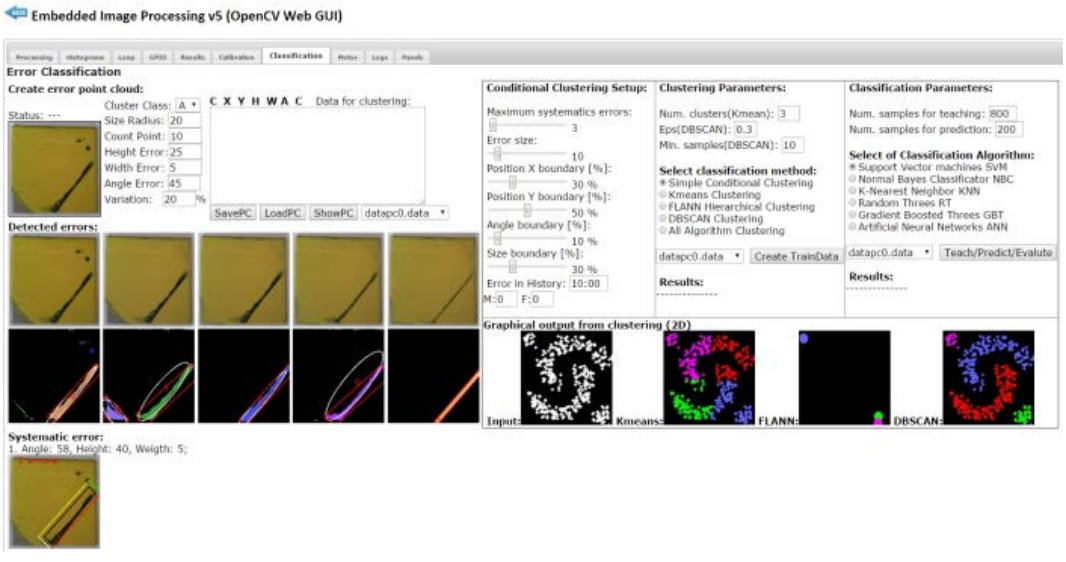

Figure 7. GUI for error parameter extraction by image processing

5. Experiments with Clustering/ Classification Algorithms

We conducted the experiment with metal square parts 35x35 mm painted by polyurethane yellow paint. Possible errors were classified to the four basic classes (A, B, C, D). These errors can arise as result of rough handling, manipulation, drying and poor surface pretreatments before painting. Examples of surface errors samples used in experimental verification are shown on the Figure 8.

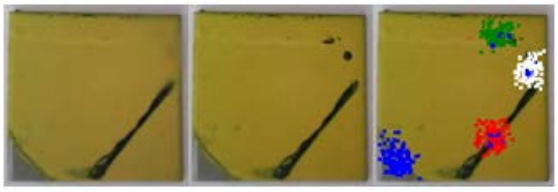

Figure 8. Samples used for experiments.

Result of clustering in DBSAN in comparison to next two algorithms is shown on the Figure 9.

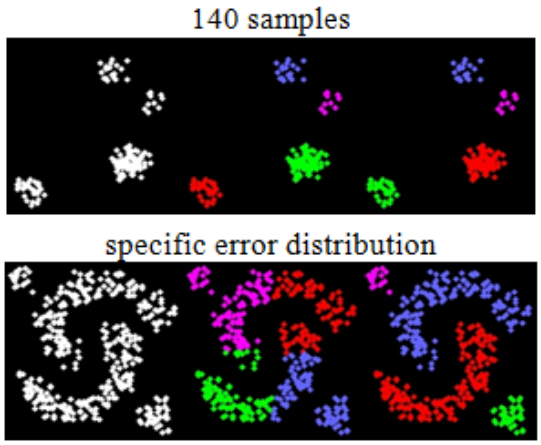

Figure 9. Picture from left: Clear data, Kmeans, DBSCAN

For comparison of result we compare reliability of MLP/ANN teaching with other classification algorithms. Graphical representation is shown on the Figure 10.

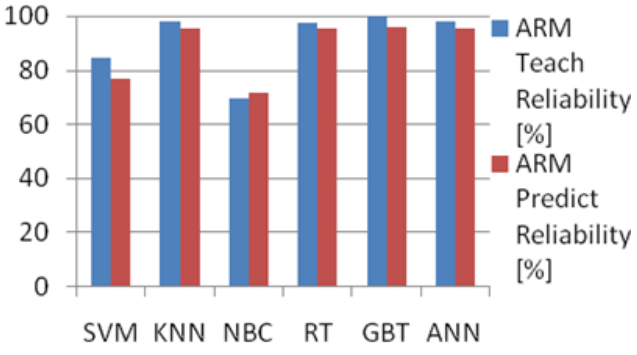

Figure 10. Teaching and prediction reliability for 1000 samples

The experiments show that the most reliable algorithm is Gradient Boosted Threes [GBT] with reliability 99,9 / 95,8 % and neural networks [ANN] 97,8 / 95,3 %. In terms of the duration of the learning model was K-Nearest Neighbor [KNN] the fastest one, achieving sufficient accuracy subsequent classification of new patterns of 97,8 / 95,2 %. In real task, the classification learning must be repeated in certain defined intervals, it requires minimum time of new samples teaching and the minimum time in class assignments with adequate reliability.

6. Conclusions

The article describes methodology of systematic error diagnostics by clustering and classification of advanced machine learning methods and algorithms. Designed software recognizes surface error, creates database of errors and classify them to error classes.

Density based clustering (DBSCAN) is ideal solution for specific class distribution, but with significantly increasing delay for large datasets. We compare reliability of DBSCAN with K-Means algorithm.

For classification we use MLP/ANN algorithm. We compare reliability with these classification algorithms: Support Vector Machines (SVM), Random Threes, Gradient Boosted Threes, K-Nearest Neighbor, Normal Bayes Classifier. ANN algorithm can be more precise after some optimization. Data results from recognition are stored in database and can be used for statistics of production process efficiency. The result of the experiment can be a different for another set of training and test data. The next works will be aimed to experiments with other advanced machine learning algorithms.

Acknowledgement

The research was supported by the Project VEGA 1/0911/14 Implementation of wireless technologies into the design of new products and services to protect human health.

References

[1] E. R. Davies, Computer & Machine Vision, *Theory Algorithms Practicalities*, Elsevier, p. 934.

[2] S. Har-Peled, B. Sadri, How fast is the k-means method?, *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages, Philadelphia, PA, USA, 877–885, (2005).

[3] D. Arthur, S. Vassilvitskii,K-means++: the advantages of careful seeding, *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, (2007).

[4] W. H. E. Day, H. Edelsbrunner, Efficient algorithms for agglomerative hierarchical clustering methods, *Journal of Classification*, 1,7-24, (1984).

[5] M. Muja, D. G. Lowe, Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration, *in International Conference on Computer Vision Theory and Applications*, VISAPP'09, (2009).

[6] M. Ester, H.-P. Kriegel, J. Sander, X. Xu E. Simoudis, J. Han, U. Fayyad, A density-based algorithm for discovering clusters in large spatial databases with noise, *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, vol. 1AAAI Press (1996), pp. 226-231.

[7] C. C. Chang, C.-J. Lin., LIBSVM, a library for support vector machines, *ACM Transactions on Intelligent Systems and Technology*, 2:27: 1-27:27, (2011).

[8] O. Kadri, L. H. Mouss, M. D. Mouss, Fault diagnosis of rotary kiln using SVM and binary ACO, *The Journal of Mechanical Science and Technology*, vol. 26, no. 2, pp.601-608, (2012).

[9] K. Fukunaga, Introduction to Statistical Pattern Recognition., New York: Academic Press, (1990).

[10] R.K. Agrawal, R. Bala, Incremental Bayesian classification for multivariate normal distribution data, Volume 29, Issue 13, pp.1873-1876, (2008).

[11] D. Coomans, D.L. Massart, Alternative k-nearest neighbour rules in supervised pattern recognition, k-Nearest neighbour classification by using alternative voting rules, , AnalyticaChimica, Acta 136, pp. 15-27, (1982).

[12] B. Yao, F. Li,P. Kumar, K-Nearest Neighbor Queries and KNN-Joins in Large Relational Databases (Almost) for Free, *Data Engineering (ICDE), 2010 IEEE 26th International Conference*, pp. 4-15, (2010).

[13] L. Breiman, A. Cutler, Random Forests, available at: <www.stat.berkeley.edu/~breiman/RandomForests/cc_graphics.htm>.

[14] Bo-Suk Yang, Random forests classifier for machine fault diagnosis, *The Journal of Mechanical Science and Technology*, vol. 22, no. 9, pp.1716-1725, (2008).

[15] J. Friedman, Greedy Function Aproximation, A Gradient Boosting Machine, Feb. 24, 1999, available at: <docs.salford-systems.com/GreedyFuncApproxSS.pdf>.

[16] Y. LeCun, L. Bottou, G.B. Orr and K.R. Muller, Efficient backprop, in Neural Networks, Tricks of the Trade, Springer Lecture Notes in Computer Sciences 1524, pp.5-50.

[17] K. Židek, E, Rigasová, Diagnostics of Products by Vision System, Applied Mechanics and Materials, Trans Tech Publications, Switzerland, , Vol. 308, 33-38, (2013).