Classification of Wafer Maps Defect Based on Deep Learning Methods With Small Amount of Data

Kudrov Maksim[1], Bukharov Kirill[1], Zakharov Eduard[1],
Grishin Nikita[1], Bazzaev Aleksandr[1], Lozhkina Arina[1],
Semenkin Vladislav[1], Makhotkin Daniil[1], Krivoshein Nikolay[1]

[1]Dept. of Aeromechanics and Flight Engineering
Moscow Institute of Physics and Technology (MIPT)
Zhukovsky, Russia
mkudrov@mail.ru, bukharovkd@gmail.com, eddyzaharov@mail.ru,
{grishin.na, bazzaev.af, lozhkina.as, semenkin, mahotkin.dr, krivoshein.ne}@phystech.edu

***Abstract*—This paper attempts to solve the problem of defect classification for the purpose of automation of the processing of flaw detection results. Article proposes an algorithm based on deep convolutional neural networks (DCNN) for recognizing patterns of defects in semiconductor wafers. In order to train the model, a composite training data set was created and applied. Its basis consists of synthetic data and an extra small amount of experimental data including about 20 examples. Verification of the work was carried out on an open data set WM–811K. The resulting classification accuracy is about 87.8%. This is a satisfactory result from a practical point of view. The developed algorithms can be used both in software systems for data analysis in the production of semiconductor wafers, as well as part of separate software modules for electronic flaw detectors.**

***Keywords*—convolutional neural network, deep learning, flaw detection, wafer maps, small data, pattern classification.**

I. Introduction

The production of semiconductor wafers is a complex and expensive process. Depending on the wafer diameter and chip size each wafer may contain from dozen to thousand chips. During the assembling, each wafer goes through a series of technological operations. Equipment or technological fault at any operation produces a variety of defects on the wafer. Grouping the systematic defects with similar shape and locations properties into the pattern allows to estimate the relation with the root causes of hardware failure and allows to rapidly eliminate the problem [1]. Nowadays most of semiconductor wafers manufactures do not use systems for automatic defects patterns analyzing and have to evaluate statistical information about appeared patterns manually.

In the conditions of increasing amount of manufactured semiconductor elements, tendency of downsizing chips, growing sizes of semiconductor wafers [2], extending functional possibilities of modern electronic flaw detectors, stricter requirements for output control systems, including the requirements for reducing the human factor increases the necessity for automation of the process of flaw detection. Thus, it is essential to investigate the new approaches for recognition and classification of the manufacturing defects on the semiconductor wafers.

Modern methods of classification patterns of defects in semiconductor elements are based on the machine learning methods, such as deep convolutional neural networks (DCNN) approaches [3] – [7], Support Vector Machine [8], Decision Trees Ensembles [9], Randomized General Regression Network [10] and the method based on Generative Adversarial Network [11].

According to the recent researches [3] – [7], DCNN is proved to perform the most satisfying results for the pattern classification problem. The main specific of these models is the ability to classify patterns without manual feature extraction. At the same time, they have such disadvantages as the requirements for a large amount of labeled experimental data and model retraining.

Generally, studies in this area are devoted to training DCNN models on the private source data [5] – [7]. The other part of researches is focused on training on the public data set WM–811K [3], [4]. The main problem of exploiting private dataset for training, testing and validation DCNN is the impossibility to conduct wide response from the scientific community. Public datasets, on the contrast, make it possible to develop open source solutions and validate them using different data science approaches. Limited labeled data from manufacturers raises requirement to find new approaches for small amount of data.

The purpose of the presented research was to improve the quality of pattern recognition method in conditions of a small amount of labeled experimental data.

The paper presents a method of preparing a composite training dataset, strategy of training DCNN models, validation and final testing results for different DCNN architectures and the investigated dependence of recognition accuracy to the ratio of experimental labeled data to synthetized data for the current problem.

In order to solve the problem of non-representativeness of experimental data, an approach based on the formation of a composite dataset was developed. The key feature of applying this approach is the possibility to expand relatively small amount of labeled experimental dataset (from 10 to

120 examples for each class) with synthesized data based on general manufacturers pattern descriptions.

Learning strategy for DCNN contains two steps: pre-training model on pure synthetic dataset and main training and validation on composed dataset.

The reliability of the simulation results is confirmed by isolated final testing on the WM–811K dataset.

This article has the following structure. Section II describes methods for the preparation of a composite training dataset. Section III presents the statistical parameters of the training data set. Section IV formulates learning strategies for DCNN models. Section V shows the results of numerical simulations. The conclusion is given in Section VI.

II. The Description of Data Generation

A. Defect Pattern Types

Among many patterns formed during the semiconductor wafer fabrication process, 7 patterns could be considered typical, as they are repeated in various articles [9], [12] – [16]. Descriptions of their structure and the causes of their origin are given in the Table I. The proposed definitions based on numerous studies allow us to form a concept for preparing a composite dataset which is based on a technological point of view.

TABLE I
Synthetic Defect Pattern Description

Pattern	Pattern description	Typical cause of origin
Donut	Annular clusters	Redeposition of dissolved photoresist solids backing onto the wafer surface during developing process, as the center rinsing step results less defective at the center of the wafer [13].
Scratch	Straight and curved lines	Human error at the shipping and handling process [12]. Error at chemical mechanical polishing (CMP) [14].
Loc	Localized clusters	Contiguous region of a wafer fail is caused by excess vibration in a given machine liberating enough particles [15]. Crystalline heterogeneity [12].
Center	Clusters at the center of wafers	Problem in the plasma area [15]. Thin film deposition [9].
Edge-Loc	Localized clusters on the edge of the wafer	Uneven heating during diffusion process [16].
Edge-Ring	Ring-shaped clusters around the perimeter of the wafer	Abnormal temperature control in the rapid thermal annealing process [10].
Random	One or more defects that cannot be attributed to one of the patterns	Defects of this pattern are not systemized.

The formation of the synthetic part of the composite dataset is based on a general approach; however, fine-tuning to specific production needs is also possible. The operability of offered in this paper approach is confirmed by final test results of the analysis of the open dataset WM–811K.

Example of each type pattern of WM–811K is shown in Figure 1. Table II shows the quantitative distribution of patterns in WM–811K. Those wafer map images that are larger than 40x40 pixels are selected from this dataset and resized to the size of 96x96 pixels.

TABLE II
Quantitative Pattern Distribution in WM–811K

Pattern	Center	Donut	Loc	Scratch	Edge-Ring	Edge-Loc
Percentage in sample	4.1%	3.4%	9.0%	5.0%	63.5%	15.0%
Quantity	414	348	911	503	6472	1523

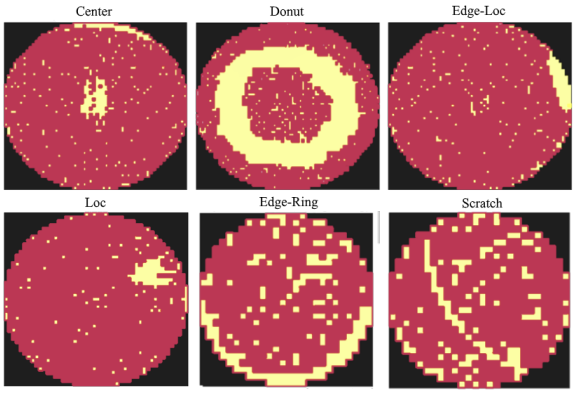

Fig. 1. WM–811K pattern examples.

B. Synthesis of Wafer Map Defect Patterns

In order to create the synthetic part of the composite dataset, parametric mathematical models were developed for each of Table I presented patterns. Descriptions of those models are given below separately.

In this paper we deliberately do not use standard data augmentation techniques. There are several reasons for that. One of the reason is that data augmentation by standard methods can improve classification results, but at the same time, the model tends to be overfitted because of the lack of manufacturing causes of defects in the methodology. It happens because of the location of some patterns may be a key characteristic of a defect.

The completeness of the presentation of the defect model compensates for the absence of augmentation. The mathematical model is flexible for defects location and local shape differences.

C. Dataset Regularization

The synthesis process of each type of defect patterns also includes a method for regularizing and visually approximating the generated wafer maps with defect patterns to experimental ones modeled using Poisson point process and a number of morphological operations. The Poisson distribution is given by:

$$f(k,\lambda) = \frac{\lambda^k \exp^k}{k!} \tag{1}$$

where f describes the probability of occurrence of k events on the interval λ. In this model, it is assumed that the probability of a small random defect does not depend on other similar defects.

D. Donut Pattern

The generation of the Donut pattern is a fundamental operation for subsequent synthesis of other patterns. Allowable range of synthesis hyperparameters consists of the central point of the pattern, the radius vector of the outer and inner borders of the rings and angular sectors. The synthesis process starts at the central point and continues between geometrical locations of the points lying on or between the outer and inner borders of the ring according to the given range of solid angles. The defect parameterization is defined as follows:

$$(x - x_0)^2 + (y - y_0)^2 = r^2, r_{in} \leq r \leq r_{out}, \tag{2}$$

where x_0, y_0 – coordinates of the central point; r_{in}, r_{out} – radius vectors of outer and inner borders of the ring respectively; x, y – coordinates of defect points.

$$\begin{cases} x = x_0 + r\cos(\alpha) \\ y = y_0 + r\sin(\alpha) \end{cases}, r_{in} \leq r \leq r_{out}, \alpha_1 \leq \alpha \leq \alpha_2, \tag{3}$$

where α_1, α_2 – angles of the beginning and end of the sector, respectively.

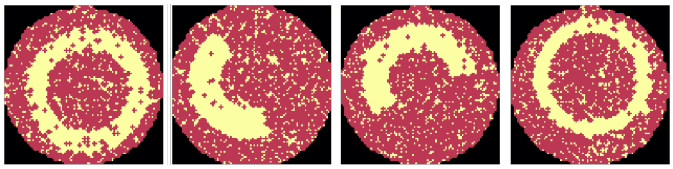

Fig. 2. Examples of generated the Donut pattern.

E. Scratch Pattern

The Scratch pattern consists of sequential synthesis processes of broken curved lines over several segments. The synthesis parameters of this defect pattern are the characteristics of each broken line: the starting point, the length of the line, the direction vector, the number of segments. System of equations is formed based on the given parameters. The solution of this equations is the original pattern for each line of the following:

$$\begin{cases} x^i = p_1^i t^i + x_0^i \\ y^i = p_2^i t^i + y_0^i \end{cases}, \tag{4}$$

where p_1, p_2 – direction vectors; t – real number; x_0, y_0 – coordinates of the starting point, i – segment index.

F. Loc Pattern

The Loc pattern inherits the synthesis architecture of the Donut pattern. Synthesis parameters are the angular size of the sector, the starting points and the radius vector of the outer and inner borders of the rings. The parameterization of the pattern is:

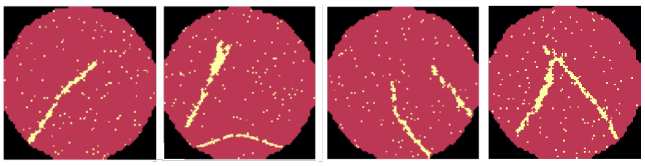

Fig. 3. Examples of generated the Scratch pattern.

$$\begin{cases} x = x_n + r\cos(\alpha) \\ y = y_n + r\sin(\alpha) \end{cases}, r_{in} \leq r \leq r_{out}, \alpha_1 \leq \alpha \leq \alpha_2, \tag{5}$$

where x_n, y_n – the coordinates of the starting points.

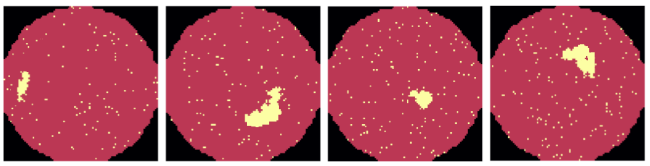

Fig. 4. Examples of generated the Loc pattern.

G. Center Pattern

The Center pattern also inherits the Donut pattern build architecture. However, the generation of this defect sets limits on the allowable range of parameters of the mathematical model in comparison with the Loc pattern Eq. (5).

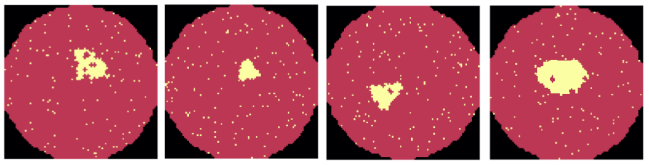

Fig. 5. Examples of generated the Center pattern.

H. Edge-Loc Pattern

The process of synthesis of the Edge-Loc pattern is similar to the technique for the Loc pattern Eq. (5) and is obtained by limiting the range of allowable parameters of the mathematical model.

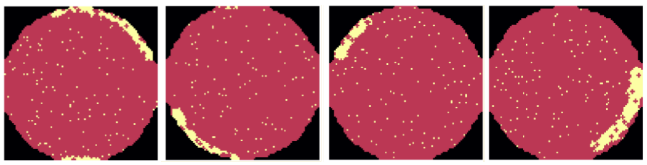

Fig. 6. Examples of generated the Edge-Loc pattern.

I. Edge-Ring Pattern

The Edge-Ring pattern has a similar build model to Eq. (2) and Eq. (3) of the Donut pattern. Synthesis parameters are limited in terms of the central point, the external and internal radii.

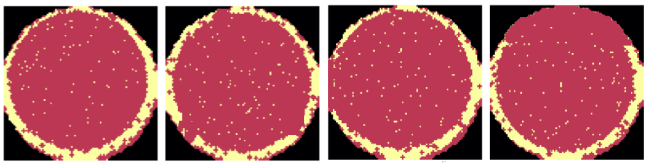

Fig. 7. Examples of generated the Edge-Ring pattern.

III. Training Dataset Structure

Training composite dataset consists of experimentally obtained and artificially synthesized data which contains 14000 generated wafer maps for seven classes. Real wafer maps for the composite set are randomly selected in the amount of 100 examples of each class for the main learning. Thus, the ratio of experimentally obtained data to the synthesized data in the training data set is 1 to 20 or briefly $R_{ls} = 0.05$.

The composite data set is divided into training and verification in the ratio of 80% and 20%. The final verification of the method is carried out on the remaining experimental data that didn't fall into the composite dataset.

IV. Learning Strategy

To increase speed of convergence and accuracy of a model we use pretrained weights. In this case before main train stage on composited dataset each model is originally trained and validated on a dataset consisting only of synthesized data.

The neural network is trained via the backpropagation method. During the training, an adaptive assessment of the learning parameters is carried out using the stochastic gradient descent algorithm. Learning speed is changed by the cosine rule. The number of training epochs are chosen equal to 50. Cross-validation for random samples is used as a method to combat overfitting of the model, the number of cross-validation samples are taken to be 5. Categorical cross entropy is used as a metric for the quality of classification.

The final test consists of only WM-811K examples.

V. Experiment and Results

The proposed method for solving the problem of classifying wafer maps patterns is implemented in Python using the OpenCV and PyTorch open libraries. We use a platform with an Intel Core i5-8600 @ 3.10 GHz processor (6 cores), 32 GB memory and NVIDIA GeForce 1080 GPU.

During our research we analyzed neural networks with the following architectures: VGG, ResNet and MobileNet. The main task of comparing the models is to evaluate the performance and accuracy. We use Accuracy, Recall, Precision and F1 score to evaluate the recognition accuracy.

Table III shows the learning outcomes of various neural network architectures and their training time.

To obtain a test accuracy of 87.8% for testing the ResNet-50 on the open dataset WM–811K, it is enough that $R_{ls} = 0.05$. The corresponding confusion matrix for the ResNet-50 for six patterns is demonstrated in Figure 8. Table IV shows the classification results for various patterns.

TABLE III
Comparison of the Results of DCNN ($R_{ls} = 0.05$)

Model name	**Train Accuracy**	**Final Test Accuracy (not pretrained)**	**Final Test Accuracy (pre-trained)**	**Performance [Img/sec]**
Dataset	**Composed**	**WM-811K**		
VGG19	0.9946	0.8326	0.8481	49.2
ResNet34	0.9951	0.8274	0.8539	84.1
ResNet50	**0.9951**	**0.8625**	**0.8784**	**58.4**
MobileNetV2	0.9939	0.7822	0.8191	66.8

Example of classification of wafer map defects along with the distribution of the probabilities are displayed in Figure 9.

It is possible to build a dependence of accuracy DCNN models to different values of R_{ls}. These dependencies are shown in Figure 10 for each pretrained DCNN model.

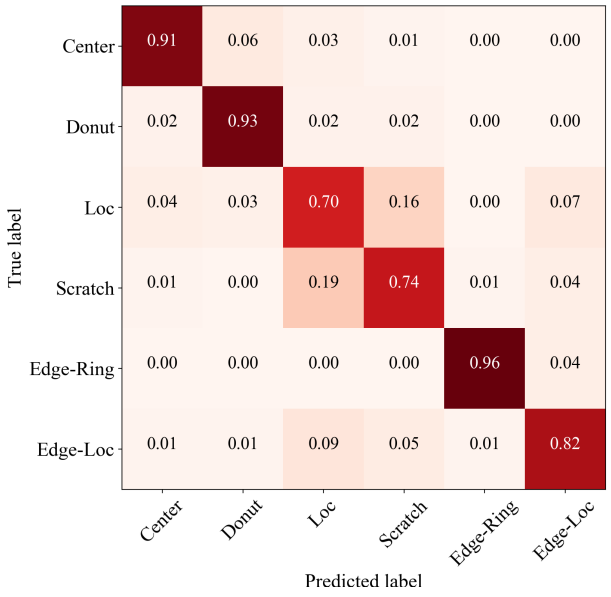

Fig. 8. Normalized confusion matrix for ResNet-50 ($R_{ls} = 0.05$).

TABLE IV
Classification Results (ResNet-50, $R_{ls} = 0.05$)

Pattern name	**Accuracy**	**Precision**	**Recall**	**F1 score**
Center	0.9131	0.7837	0.9236	0.8479
Donut	0.9328	0.8287	0.8589	0.8436
Loc	0.7170	0.7380	0.6078	0.6667
Scratch	0.7421	0.4677	0.8635	0.6068
Edge-Ring	0.9601	0.9878	0.9412	0.9639
Edge-Loc	0.8273	0.7296	0.7491	0.7401

VI. Conclusion

The paper presents the results of research devoted to a pattern classification problem in conditions of a small amount of labeled experimental data. We have proposed a new method of

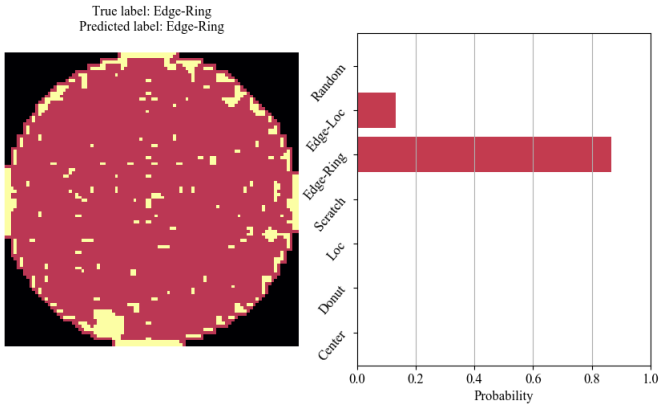

Fig. 9. Example of the ResNet-50 ($R_{ls} = 0.05$) result classification.

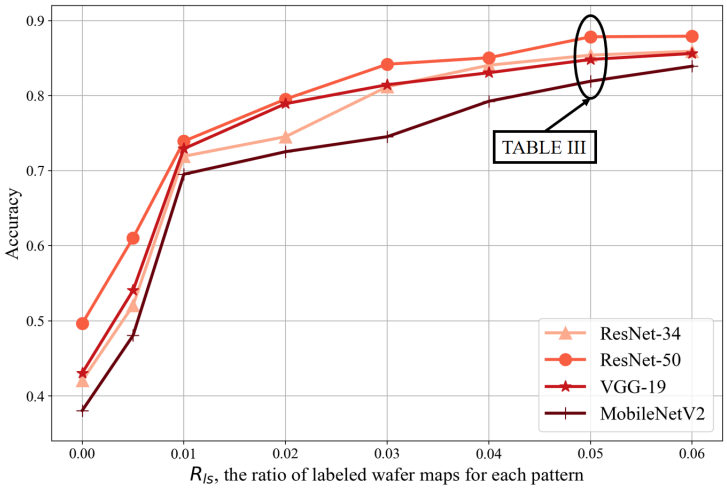

Fig. 10. The dependence of the models accuracy to the ratio of labeled data.

preparing a composite training dataset based on general pattern descriptions obtained from expert analysis of technological processes. The efficiency of this method is shown for different DCNN model architectures.

A new learning DCNN model strategy, which includes the pretraining step, was developed. The pretraining model on pure synthetics dataset improves the resulting accuracy by 1% up to 4% depending on the model architecture. The reliability of the simulation results is confirmed by isolated final testing on the WM-811K dataset.

During the research, the various DCNN models such as VGG-19, ResNet-34, ResNet-50, MobileNetV2 was benchmarked for a different ratio of experimental labeled data to synthesized data, R_{ls} in range from 0 to 0.06. It shows the dependence of final model accuracy on the R_{ls} value and gives an idea of the amount of data that needs to be prepared in order to obtain a required accuracy.

It is demonstrated that the leader in final classification accuracy with $R_{ls} = 0.05$ on the public dataset WM–811K is ResNet-50 model which provides 87.8% accuracy.

The developed algorithms can be used in software systems for data analysis in the production of semiconductor wafers and as part of separate software modules for multipurpose electronic flaw detectors.

In order to improve the presented results, it is possible to increase the number of synthesized subclasses of patterns, such as curved scratches. The next step of the research this approach is consisted in applying force teaching, solving multi-classification and segmentation problem on small amount of data.

REFERENCES

[1] A. Drozda-Freeman, M. McIntyre, M. Retersdorf, C. Wooten, X. Song and A. Hesse, "The Application and Use of an Automated Spatial Pattern Recognition (SPR) System in the Identification and Solving of Yield Issues in Semiconductor Manufacturing", *2007 IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, Stresa, pp. 302-305, 2007.

[2] F. D. Palma, G. D. Nicolao, G. Miraglia, E. Pasquinetti, and F. Piccinini, "Unsupervised Spatial Pattern Classification of Electrical-Wafer-Sorting Maps in Semiconductor Manufacturing", *Pattern Recognition Letters*, vol. 26, no. 12, pp. 1857-1865, 2005.

[3] Ming-Ju Wu, Jang, J. and Jui-Long Chen, "Wafer Map Failure Pattern Recognition and Similarity Ranking for Large-Scale Data Sets", *IEEE Transactions on Semiconductor Manufacturing*, 28(1), pp.1-12.

[4] T. Ishida, I. Nitta, D. Fukuda and Y. Kanazawa, "Deep Learning-Based Wafer-Map Failure Pattern Recognition Framework", *20th International Symposium on Quality Electronic Design (ISQED)*, Santa Clara, CA, USA, pp. 291-297, 2019.

[5] S. Cheon, H. Lee, C. O. Kim and S. H. Lee, "Convolutional Neural Network for Wafer Surface Defect Classification and the Detection of Unknown Defect Class", *IEEE Transactions on Semiconductor Manufacturing*, vol. 32, no. 2, pp. 163-170, May 2019.

[6] T. Nakazawa and D. V. Kulkarni, "Wafer Map Defect Pattern Classification and Image Retrieval Using Convolutional Neural Network", *IEEE Transactions on Semiconductor Manufacturing*, vol. 31, no. 2, pp. 309-314, May 2018.

[7] Y. Yuan-Fu, "A Deep Learning Model for Identification of Defect Patterns in Semiconductor Wafer Map", *30th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC)*, Saratoga Springs, NY, USA, pp. 1-6, 2019.

[8] Mengying Fan, Qin Wang and B. van der Waal, "Wafer Defect Patterns Recognition Based on OPTICS and Multi-Label Classification", *IEEE Advanced Information Management, Communicates*, Electronic and Automation Control Conference (IMCEC), Xi'an, pp. 912-915, 2016.

[9] M. Piao, C. H. Jin, J. Y. Lee and J. Byun, "Decision Tree Ensemble-Based Wafer Map Failure Pattern Recognition Based on Radon Transform-Based Features", *IEEE Transactions on Semiconductor Manufacturing*, vol. 31, no. 2, pp. 250-257, May 2018.

[10] G. Tello, O. Y. Al-Jarrah, P. D. Yoo, Y. Al-Hammadi, S. Muhaidat and U. Lee, "Deep-Structured Machine Learning Model for the Recognition of Mixed-Defect Patterns in Semiconductor Fabrication Processes", *IEEE Transactions on Semiconductor Manufacturing*, vol. 31, no. 2, pp. 315-322, May 2018.

[11] J. Wang, Z. Yang, J. Zhang, Q. Zhang and W. K. Chien, "AdaBalGAN: An Improved Generative Adversarial Network With Imbalanced Learning for Wafer Defective Pattern Recognition", *IEEE Transactions on Semiconductor Manufacturing*, vol. 32, no. 3, pp. 310-319, Aug. 2019.

[12] Y. Jeong, S. Kim and M. K. Jeong, "Automatic Identification of Defect Patterns in Semiconductor Wafer Maps Using Spatial Correlogram and Dynamic Time Warping", *IEEE Transactions on Semiconductor Manufacturing*, vol. 21, no. 4, pp. 625-637, Nov. 2008.

[13] W. H. Ng, S. I. Yet, and C. Y. Liau, "The Effect of UPW Quality on Photolithography Defect", *Lithography Asia 2009*, Mar. 2009.

[14] Pan T. et al. "Artificial Intelligent Matching for Scratches of Semiconductor Wafers Based on a K-NN Algorithm", *Surface Topography: Metrology and Properties*, vol. 7, no. 2, 2019

[15] C. Hansen and P. Thyregod, "Use of Wafer Maps in Integrated Circuit Manufacturing", *Microelectronics Reliability*, vol. 38, no. 6-8, pp. 1155-1164, 1998.

[16] M. H. Hansen, V. N. Nair, and D. J. Friedman, "Monitoring Wafer Map Data from Integrated Circuit Fabrication Processes for Spatially Clustered Defects", *Technometrics*, vol. 39, no. 3, p. 241, 1997.