



Development of taxonomy for classifying defect patterns on wafer bin map using Bin2Vec and clustering methods

Dong-Hee Lee ^{a,*¹}, Eun-Su Kim ^b, Seung-Hyun Choi ^c, Young-Mok Bae ^b, Jong-Bum Park ^b, Young-Chan Oh ^b, Kwang-Jae Kim ^c

^a Department of Industrial Engineering, Sungkyunkwan University, the Republic of Korea

^b Department of NAND SE, SK hynix, the Republic of Korea

^c Department of Industrial and Management Engineering, Pohang University of Science and Technology, the Republic of Korea



ARTICLE INFO

Keywords:

Wafer bin map
Word2Vec
Bin2Vec
Defect-pattern classification
Convolutional neural networks

ABSTRACT

A wafer consists of several chips, and serial electrical tests are conducted for each chip to investigate whether the chip is defective. A bin indicates the test results for each chip with information on which tests the chip failed. A wafer bin map (WBM) shows the locations and bins of the defects on the wafer. WBMs showing spatial patterns of defects usually result from assignable causes in the wafer fabrication process; hence, they should be classified in advance. The existing defect-pattern taxonomies do not consider bins, although useful information can be obtained from them. We propose a taxonomy that consists of the shape, size, location, and bin dimensions. The bin dimension is developed using Bin2Vec method, which determines RGB (red-green-blue) code for each bin according to the spatial similarity between bins. Three levels of the bin dimension are defined by analyzing a large number of WBMs using Bin2Vec and clustering methods. Compared with the existing taxonomies, the proposed taxonomy has the advantage of identifying major bins of defect patterns, new defect patterns, and non-critical defect patterns. A high-quality training dataset was obtained using the proposed taxonomy; consequently, a defect pattern classification model with satisfactory classification performance could be obtained.

1. Introduction

The semiconductor manufacturing process comprises several hundred process steps for the fabrication of wafers. After fabrication, serial electrical probe tests are conducted for each semiconductor chip on a wafer to investigate whether the chip is defective. The chip is a physical object that exists on a wafer and bin is a variable indicating on which test the chip failed. Suppose that 10 serial electrical probe tests are conducted for each chip. The result of each electrical probe test is recorded as binary indicating that the chip is functional or defective. After the serial electrical probe tests, each chip has its own bin value according to its test results. Because there are 10 binary tests, a total of 1024 ($= 2^{10}$) bins are possible. The probe test results are represented by a wafer bin map (WBM), which is an image showing the bins and locations of chips on a wafer. A chip is a physical object that exists on wafer and bin is a variable indicating on which test the chip failed. In other words, chip itself has no information. Instead, its bin value indicates on which test the chip failed. When the wafers have same number of chips, the chip

location on the wafers is also same. However, bin distributions on wafers can be different because chips of the wafers can have different bin values.

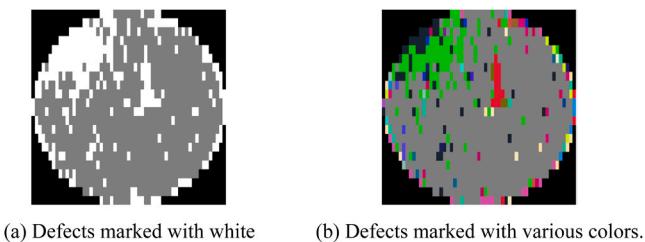
Fig. 1(a) shows an example of a WBM marked with black for the background, gray for the normal chips, and white for the defective chips (referred to as defects hereinafter). Fig. 1(b) provides more information, as each chip is marked with color according to its bin. In Fig. 1(b), two spatial patterns of the defects (referred to as defect patterns hereinafter) in the WBM can be identified: 1) green defects are clustered on the upper-left part of the WBM, and 2) red defects are placed vertically, exhibiting a scratch pattern.

Defect patterns are usually due to assignable causes in the fabrication process, such as machine abnormalities and operator errors. They can be prevented by identifying and eliminating the assignable causes. The first task for identifying the assignable causes is detecting wafers with defect patterns and classifying the patterns. Recently, automatic classification of defect patterns has been highlighted because of its advantages of low cost, full inspection, and good classification performance. Various

* Corresponding author.

E-mail address: dhee@skku.edu (D.-H. Lee).

¹ ORCID: 0000-0001-8549-8992.



(a) Defects marked with white (b) Defects marked with various colors.

Fig. 1. Example of WBMs.

methodologies have been applied for automatic classification, such as convolutional neural networks (CNNs), decision trees, regression analyses, and support vector machines (Yu and Lu, 2016; Piao et al., 2018; Adly et al., 2015; Fan et al., 2016; Nakazawa and Kulkarni, 2018; Kyeong and Kim, 2018; Saqlain et al., 2019; Shim et al., 2020). Because these are supervised learning methods, training data should be prepared by labeling WBMs. For coherent labeling, defect patterns should be defined in advance. For example, the WM811K taxonomy is a representative method for defining defect patterns. Defect patterns observed in the WM811K dataset—the largest WBM dataset available to the public (Wu et al., 2015)—are classified according to the WM811K taxonomy, as shown in Fig. 2.

Extensive studies have been conducted to define various defect patterns. However, the existing defect-pattern taxonomies, including the WM811K taxonomy, have common limitations in that they do not allow clear differentiation among defect patterns. To cope with this difficulty, Choi et al. (2021) proposed a spatial dimension-based taxonomy (SDT) with three spatial dimensions of a defect pattern: shape, size, and location. Levels are defined in each dimension with detailed criteria by reflecting the spatial features of the defect pattern; thus, the SDT can define the defect patterns clearly and differentiate them accurately. In addition, a sufficient set of patterns can be defined by properly combining the levels of the three dimensions. Later, Kim et al. (2021) employed the SDT to develop a CNN model for the defect pattern. A high-quality training dataset was obtained by coherently labeling the detected patterns according to the SDT.

Despite the advantages of the SDT, it may not be effective for identifying significant defect patterns, because it does not consider the bins of chips on WBMs, although useful information can be obtained by considering it. First, it is possible to identify a major bin in a WBM. When the defects are marked only with white regardless of their bins, as shown in Fig. 1, wafers may have similar spatial defect patterns, but the major bins constituting the patterns may not be the same. Accordingly, the assignable causes resulting in the bins may not be the same. Second, it is possible to identify the spatial distribution of each bin in the WBM; thus, new defect patterns can be defined. By tracking wafers with new defect patterns, new assignable causes can be identified.

One simple approach for considering the bin in the WBM is to apply a

random coloring approach, as shown in Fig. 1(b). In this approach, the color for each bin is randomly determined, and each chip is marked with a random color according to its bin. In such a case, the relationships among the bins cannot be preserved. Recently, Kim et al. (2019) proposed Bin2Vec—a neural network-based bin coloring method. Bin2Vec uses Word2Vec to transform the bins into three-dimensional vectors so that the spatial similarity between bins can be measured. By regarding the three-dimensional vectors as RGB codes, chips can be marked with colors according to the RGB codes. This allows similar bins to be represented by similar colors. In the probe test, if two chips are physically close to each other, their test results are likely to be similar. This is because wafer fabrication is a batch process, and nearby chips are likely to be processed under the same manufacturing conditions. The Bin2Vec-based coloring method reflects the nature of the probe test by representing similar bins with similar colors. It makes major bins, whose degree of similarity to the bin of the normal chip is low, more noticeable because most of the chips are normal. This is useful for identifying significant defect patterns.

Because of this advantage, we used Bin2Vec to develop a taxonomy that considers the bins of the defect patterns. The proposed taxonomy was developed by adding a bin dimension to the SDT; i.e., the proposed taxonomy consists of the shape, size, location, and bin dimension. Three levels of the bin dimension were defined by analyzing a large amount of WBM data using Bin2Vec and clustering methods. A total of 24 defect patterns were defined in the proposed taxonomy, and a CNN model for defect classification was developed to validate the proposed taxonomy. A high-quality training dataset was obtained using the proposed taxonomy; thus, the CNN model exhibited satisfactory classification performance. A semiconductor manufacturing company located in Korea provided the probe-test data and confirmed the effectiveness of the proposed taxonomy.

The remainder of this paper is organized as follows. Section 2 reviews the existing taxonomies for defect-pattern classification, including the SDT and Bin2Vec, which are related to the proposed taxonomy. Sections 3 and 4 explain the development and verification processes of the proposed taxonomy, respectively. Finally, the concluding remarks are presented in Section 5.

2. Literature review

2.1. Existing taxonomies for defect-pattern classification

Several taxonomies have been proposed for defect-pattern classification (Cunningham and MacKinnon, 1998). defined a single defect pattern: *Scratch* (Wang et al., 2006). considered three defect patterns: *Zone*, *Line*, and *Ring*. Here, *Line* corresponds to *Scratch* in the work of (Cunningham and MacKinnon, 1998). Other researchers proposed similar taxonomies (Hsieh and Chen, 2004; Hwang and Kuo, 2007; Yuan and Kuo, 2007; Wang, 2009; Zhou et al., 2010; Yuan et al., 2011; Choi et al., 2012; Nakata et al., 2017; Yu et al., 2021; Liu and Chien, 2013).

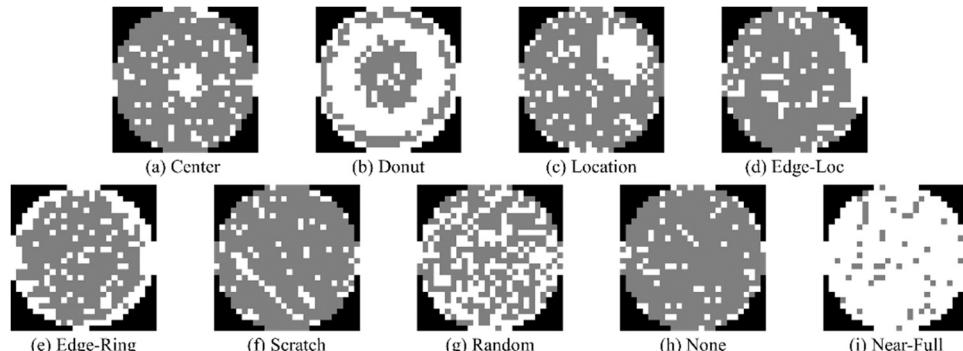


Fig. 2. WM811K taxonomy.

Table 1

Defect patterns of SDT (reprinted from Choi et al. (2021)).

Defect pattern (<i>Shape–Size–Location</i>)	Sample WBM	Defect pattern (<i>Shape–Size–Location</i>)	Sample WBM
<i>Cluster–All–Center</i>		<i>Noise–Frequent–Scattered</i>	
<i>Cluster–Big–Edge</i>		<i>Noise–Infrequent–Scattered</i>	
<i>Cluster–Small–Edge</i>		<i>Ring–All–Edge</i>	
<i>Cluster–Big–Others</i>		<i>Scratch–Long–All</i>	
<i>Cluster–Small–Others</i>		<i>Scratch–Short–All</i>	

Later, more diverse defect patterns were defined (Li and Huang, 2009; Chien et al., 2013; Liukkonen and Hiltunen, 2018; Chen and Liu, 2000).

Although such taxonomies have been widely used, their usefulness is limited by the unclear criteria for differentiation between defect patterns. The set of defect patterns is provided without a precise definition of each defect pattern. For example, in the WM811K taxonomy, the *center* refers to a defect pattern clustered in the center region, regardless of its shape. Hence, classifying a given WBM as one of the defect patterns frequently involves a high degree of subjective judgment, which hinders coherent classification.

Recently, Choi et al. (2021) proposed an SDT in which defect patterns were defined according to the *shape*, *size*, and *location* dimensions. According to the three spatial dimensions, the SDT provided a clear definition of the defect patterns. Ten defect patterns were derived using the SDT, as shown in Table 1. Nine patterns were defined (with the exception of *noise–infrequent–scattered*, which was defined for normal wafers). WBMs were labeled coherently according to the 10 detected patterns, resulting in good classification performance for the 10 defect patterns. The 10 defect patterns were validated through an investigation of the assignable causes by collaborating with the semiconductor manufacturing company.

Despite the advantage of the SDT, it has the disadvantage that the bin information is not considered. As mentioned previously, identifying major bins is important for identifying assignable causes. In addition, new defect patterns can be defined by considering the spatial distribution of bins in the WBM, which is useful for identifying the assignable causes. In this regard, we propose a new four-dimensional taxonomy that extends the SDT by adding the bin dimension. In the proposed taxonomy, Bin2Vec and clustering methods are used to develop the bin dimension. Bin2Vec is reviewed in Section 2.2.

2.2. Bin2Vec

The random RGB coloring method is a simple approach for presenting bin information in the WBM. WBMs are colorized by assigning randomly selected RGB codes to each bin so that different bin values can be distinguished in the WBM. The main drawback of this method is that the relationships between the bins are not preserved. In practice, some

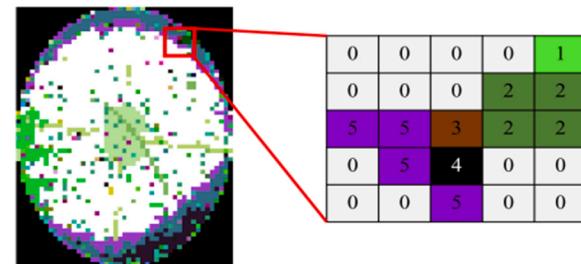


Fig. 3. Example of a target chip (brown color) and surrounding chips (reprinted from Kim et al. (2019))

bins are closely related, whereas others are completely unrelated (Choi et al., 2021).

Bin2Vec is a neural network-based bin coloring method that assigns an RGB code to each bin so that similar bins have similar RGB codes. This idea was inspired by the domain knowledge of semiconductor practitioners that nearby chips often fail the same test. Bin2Vec transforms a one-dimensional scalar bin into a three-dimensional embedding vector by employing the learning mechanism of the Word2Vec embedding model. Word2Vec is a widely used neural network-based distributed representation method that represents a word in a fixed dimension (Mikolov et al., 2013a, 2013b). Compared with traditional one-hot encoding methods, Word2Vec can preserve the semantic relationship between words. When considering words as bins, Bin2Vec can preserve the spatial relationship between the bins. Once the bin is converted into a three-dimensional vector, it can be used not only to assign an RGB code, which can also be considered a three-dimensional vector, but also to compute the similarity between any two bins (Kim et al., 2019).

There are two Word2Vec structures: continuous bag-of-words (CBOW) and Skip-Gram, which have exactly reversed structures. CBOW takes the neighborhood words as the input and is trained to correctly predict the targeted word, whereas Skip-Gram takes the targeted word as the input of the network and is trained to correctly predict the neighborhood words. Bin2Vec employs the Skip-Gram structure because it is more effective than CBOW for learning infrequent words.

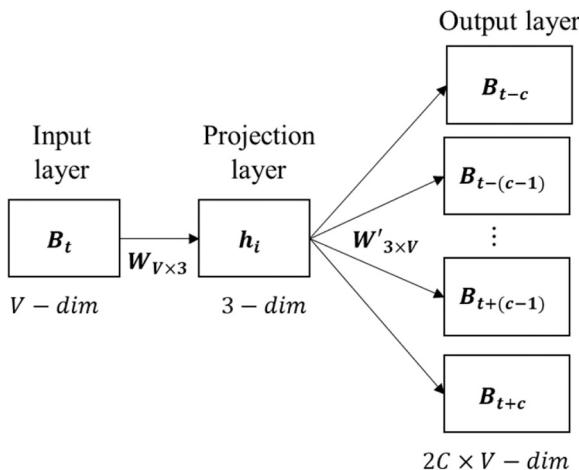


Fig. 4. Skip-Gram model architecture employed in Bin2Vec ($N = 3$).

To employ Skip-Gram, Bin2Vec defines a target chip and its surrounding chips. Fig. 3 shows an example of a target chip and its 24 surrounding chips. The numbers in Fig. 3 indicate the bins of the chips. The number of surrounding chips for each target chip is assumed to be $2C$. For example, C is set as $C = 12$ in Fig. 3. Bin2Vec designates every chip in the WBM as the target chip and its surrounding chips to train the Skip-Gram model. Fig. 4 shows the Skip-Gram structure employed in Bin2Vec (Mikolov et al., 2013a). It consists of input, projection, and output layers. The input layer takes the one-hot encoding vector of the bin of the target chip, whereas the output layer takes the one-hot encoding vectors of the bins of the surrounding chips. Suppose that we have T chips to train the Skip-Gram model, and V bins are observed in the T chips. Then, each target chip defines a single V -dimensional one-hot encoding vector as the input and $2C$ V -dimensional one-hot encoding vectors as the output according to the bins of the target and surrounding chips.

Once the V -dimensional one-hot encoding vectors are prepared, the projection layer is trained. The projection layer consists of three hidden nodes; thus, the hidden weight matrix W between the input and projection layers has dimensions of $V \times 3$, and the hidden weight matrix W' between the projection and output layers has dimensions of $3 \times V$. W and W' are trained to maximize the probability of generating the actual bins of the surrounding chips for a given bin of the target chip. Specifically, given a sequence of training chip bins (B_1, B_2, \dots, B_T), the objective function of the Skip-Gram model maximizes the average log probability, as shown in Eq. (1):

$$\underset{W, W'}{\text{maximize}} \quad \frac{1}{T} \sum_{t=1}^T \sum_{-C \leq j \leq C, j \neq 0} \log p(B_{t+j} | B_t). \quad (1)$$

The probability of generating the bins of the surrounding chips (denoted as s) given the bin of the target chip (denoted as t) is computed using a softmax function, as follows:

$$\log p(s|t) = \frac{\exp(u_s^T v_t)}{\sum_{w=1}^V \exp(u_w^T v_t)}, \quad (2)$$

where u is the column vector of the hidden weight matrix W' , and v is the column vector of the hidden weight matrix W . After the training process is complete, either the row vector of W or the column vector of W' can be used as a bin vector that serves as RGB codes.

3. Development of proposed taxonomy

The proposed taxonomy is developed through five steps: 1) data collection, 2) data preprocessing, 3) deriving RGB codes for bins using

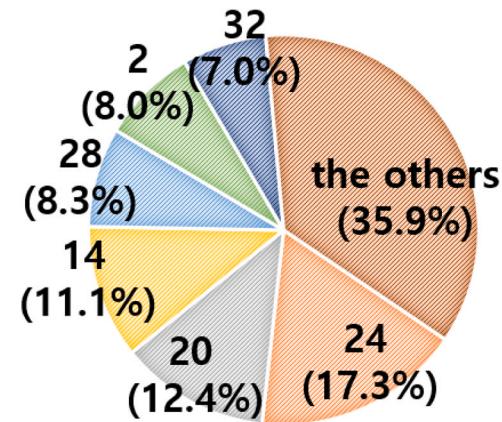


Fig. 5. Bin distribution of defects.

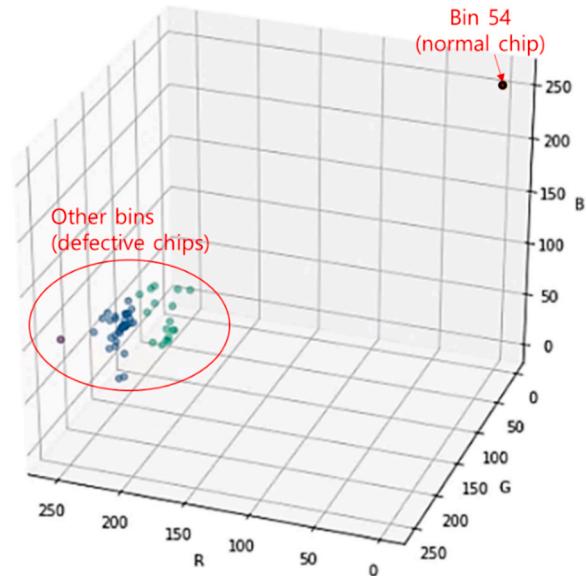


Fig. 6. Distribution of bins in the RGB space without wafer screening.

Bin2Vec, 4) clustering bins according to the RGB codes, and 5) extending the SDT by adding the bin dimension. The five steps are explained in detail in this section.

3.1. Step 1: data collection

A semiconductor manufacturing company located in Korea provided probe-test data of 10174 wafers grouped into 500 lots. The dataset is different from that of Choi et al. (2021). The dataset of Choi et al. (2021) has WBM data where only binary information for each chip, i.e., whether each chip is defective or not, exists. No bin information exists in the dataset of Choi et al. (2021). In contrast, the newly provided dataset included probe-test results for every chip on the wafers. The location of each chip on a wafer was denoted by x and y coordinates, and the bin for the chip was included. The wafer size was 27×22 , and each chip on the wafer had a bin value ranging from 1 to 57. The chips with a bin value of 54 were normal chips, whereas those with other bin values were defects. Fig. 5 shows the bin distribution of the defects. Each defect corresponded to one bin value ranging from 1 to 57 (with the exception of 54). The top six bins having large proportions in Fig. 5 (bins 24, 20, 14, 28, 2, and 32) were the major bins, and 64.1 % of the defects corresponded to one of these six bins.

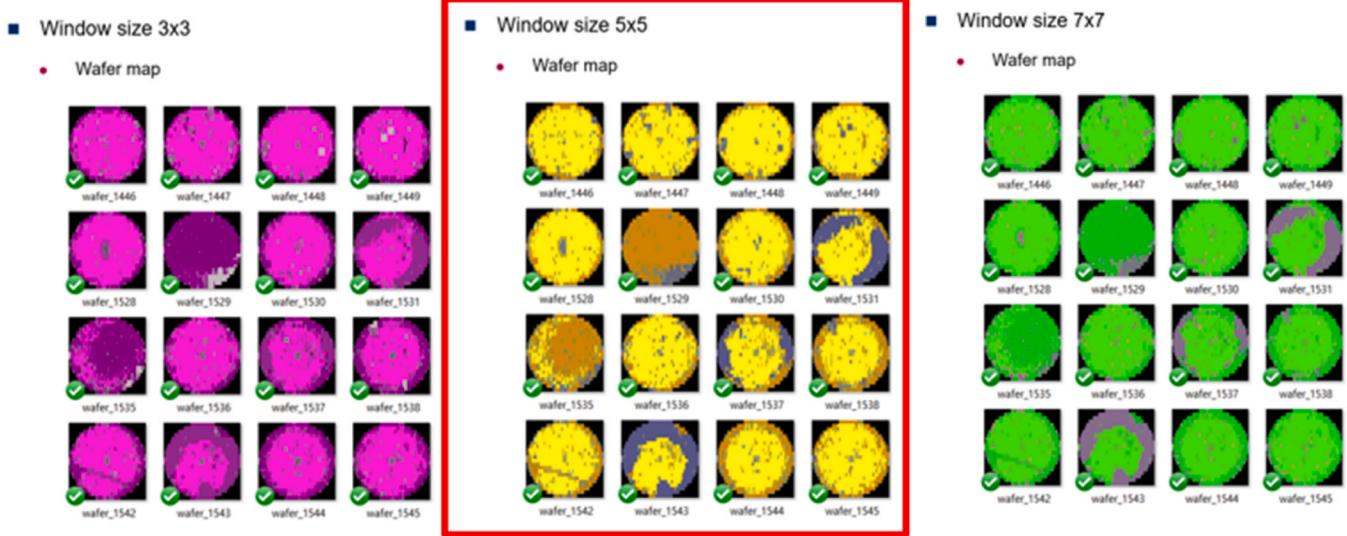


Fig. 7. Examples of WBM from three C values (left: C = 4; center: C = 12; right: C = 24).

3.2. Step 2: data preprocessing

Three types of preprocessing were conducted for the probe-test dataset. First, the wafers were screened to balance the number of wafers. Because most of the chips were normal, the training data were imbalanced. When such data are used for training the Bin2Vec model, the relationship between bins can be misleading. To investigate this issue, we trained the Bin2Vec model without wafer screening. Fig. 6 shows the RGB codes of the 57 bins obtained from the Bin2Vec model. Bin 54 exhibited an isolated RGB code, and the other bins were relatively closely clustered. When most of the chips were normal, the chips surrounding the normal chips were also likely to be normal. Because defective chips rarely exist, the defective chips can be considered as outliers in terms of the normal chip. In contrast, although majority is a normal chip, the chips surrounding the defective chip can be another defective chip. This is because the defective chips often locate nearby as shown in Table 1. This makes the RGB code of the normal chip (i.e., Bin 54) is quite different from those of defective chips. Also, this makes defective chips have similar RGB codes regardless of their bin. Consequently, it is difficult to identify the major bins.

To resolve this issue, we classified each wafer into one of the 10 defect patterns of the SDT, as shown in Table 1, and selected the wafers based on the classification result so that training data were balanced. A total of 10,174 wafers were manually classified, and 4672 wafers were found to have defect patterns. Among them, we selected 3630 wafers with six large or long defect patterns. Specifically, the six large or long defect patterns were *cluster-big-center*, *cluster-big-edge*, *cluster-big-others*, *noise-frequent-scattered*, *ring-all-edge*, and *scratch-long-all*. We selected wafers with these defect patterns because they were considered to be more critical than the other patterns. For normal wafers (i.e., wafers with a *Noise-Infrequent-Scattered* pattern in Table 1), we selected 500 wafers from 500 lots (a single wafer from each lot) to accurately represent the characteristics of the entire lot, because the wafers in the same lot are likely to be homogeneous in a batch process such as the wafer fabrication process. As a result, we selected 3630 wafers from the six large or long defect patterns and 500 wafers from the normal pattern. Each pattern commonly includes approximately 500–700 wafers; thus, the training data were balanced.

Second, pairs of bin vectors were prepared to train the Bin2Vec model. Fig. 3 shows an example of the target chip and its surrounding chips. According to the Skip-Gram structure, the bin of the target chip and the bins of the surrounding chips served as the input and output vectors, respectively. Like the example in Fig. 3, it should be decided

which chips to include as the surrounding chips. The surrounding chips can be selected by setting C value. When C changes, the training dataset changes accordingly. This results in changes in RGB of each bin; thus, the C value should be carefully selected. Basic rule for setting the value of C is considering the degree of similarity in electrical characteristics between the two chips (i.e., target and surrounding chips). The closer the chip is, the more similar the electrical characteristics to the target chip, so it should be selected as the surrounding chip. Also, larger C value generates more pairs of surrounding and target chips. This requires more computational burden for training Bin2Vec model.

We interviewed the probe test engineer about the similarity in the electrical characteristics of the two chips to determine the C value. As a result, we found that chips within a 5 × 5 grid might have high similarity in the electrical characteristics. For this reason, we set C = 12. In addition, we conducted a sensitivity analysis with respect to C to support the engineer's decision. We applied C = 4, 12, 24. As shown in Fig. 3, C equals to 12 when the grid is 5 × 5. Similarly, C equals to 4 and 24 when the grid is 3 × 3 and 7 × 7, respectively. Fig. 7 shows examples of WBM colored by Bin2Vec for the three C values. As we expected, RGBs were different according to the C values. Among them, differences in colors of bins were biggest when C = 12. This big difference in colors is helpful in defining the bin dimension. Based on the process engineer's decision and the result of the sensitivity analysis, we set C = 12.

Third, the pairs of bin vectors were deleted. When a chip and its surrounding chips had the same bin value, the pairs of bin vectors generated from these chips had the same input and output vectors. These pairs of bin vectors were deleted because they provided no information regarding the spatial similarity between bins. Additionally, when the surrounding area of a normal chip only consisted of other normal chips or non-wafer parts, the generated vectors were deleted, for the same reason. After these bin vectors were deleted, approximately 85,000,000 pairs of bin vectors remained.

3.3. Step 3: deriving RGB codes for bins using Bin2Vec

The 85,000,000 pairs of bin vectors prepared in Step 2 were transformed into 57-dimensional one-hot encoding vectors to reflect all bin values. Each pair of bin vectors consisted of two 57-dimensional vectors: one for the input vector and the other for the output vector. The Bin2Vec model was trained using 85,000,000 pairs of one-hot encoding bin vectors. We set the number of epochs as 100, which is a common value. After the training was complete, a three-dimensional vector representing the RGB code was obtained for each bin. Additionally, we conducted

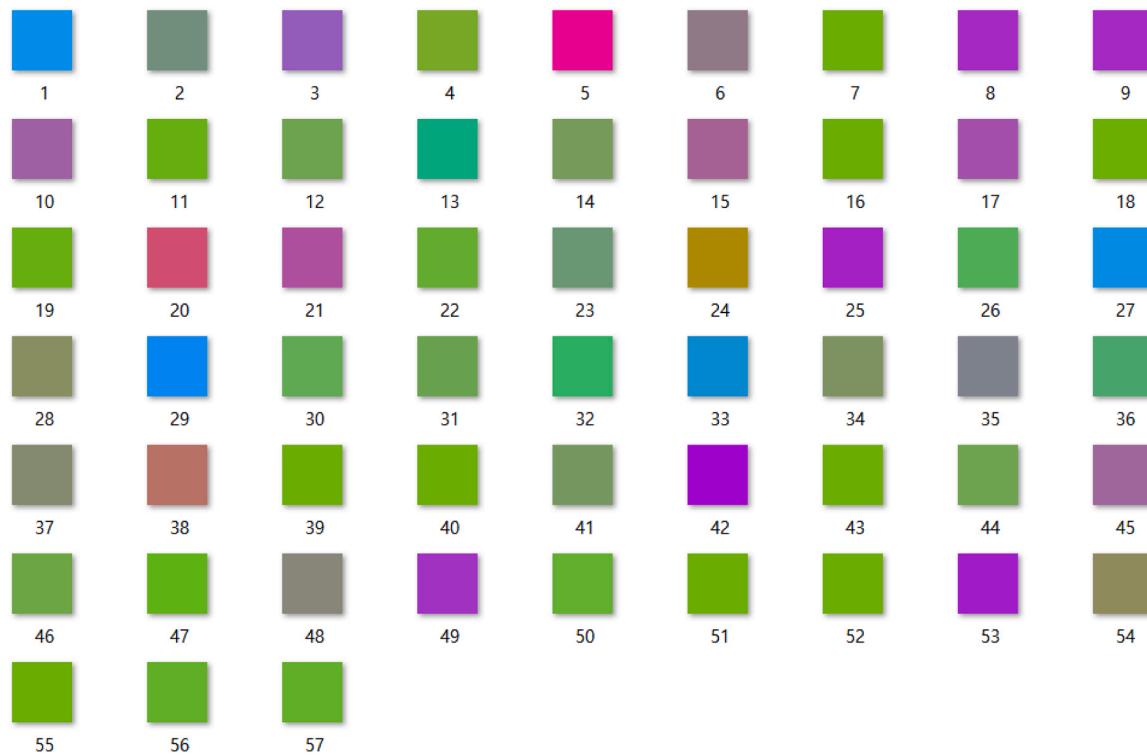


Fig. 8. RGB colors derived using Bin2Vec.

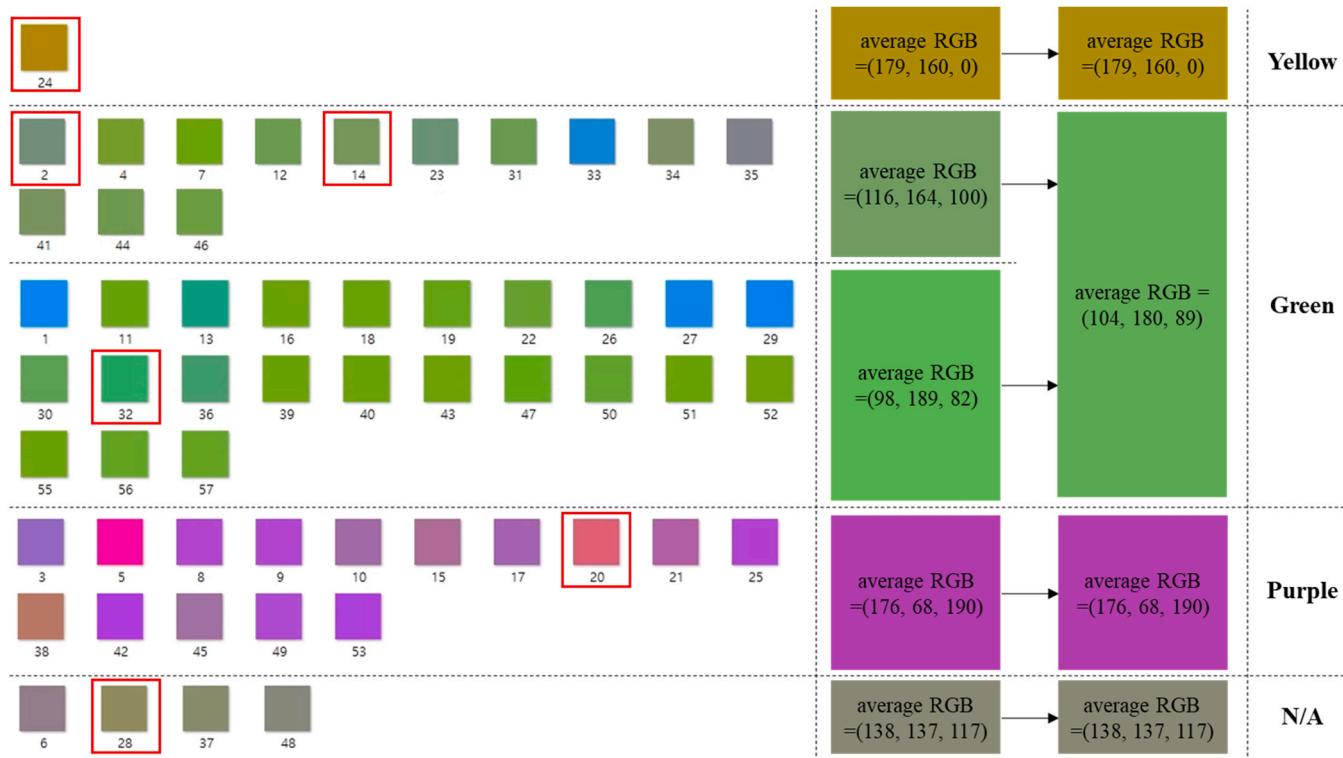
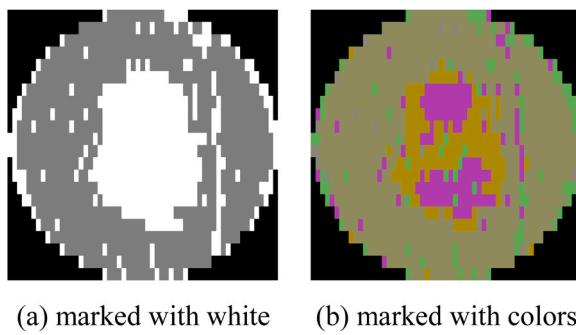


Fig. 9. Results of the bin clustering.

min-max normalization for the three-dimensional vectors to make the vectors have values between 0 and 1 and then transformed these values again by multiplying by 255 to represent RGB codes, as suggested by Kim et al. (2019). Fig. 8 shows the RGB colors for the 57 bins.

3.4. Step 4: clustering bins according to RGB codes

In Step 3, 57 RGB codes were derived, as shown in Fig. 8. These RGB codes are used to define a new dimension—the bin dimension—and its levels for extending the SDT. There are issues to be considered when



(a) marked with white (b) marked with colors

Fig. 10. Example of the “8” defect pattern.

Table 2
The 24 defect patterns defined in the proposed taxonomy.

Defect pattern (Shape–Size– Location–Bin)	Sample WBM	Defect pattern (Shape–Size– Location–Bin)	Sample WBM	Defect pattern (Shape–Size– Location–Bin)	Sample WBM
1. Cluster– Big–Center– Yellow		9. Cluster–Big– others–Purple		17. Noise– Frequent– Scattered– Green	
2. Cluster– Big–Center– Green		10. Ring–All– Edge–Yellow		18. Noise– Frequent– Scattered– Purple	
3. Cluster– Big–Center– Purple		11. Ring–All– Edge–Green		19. 8–All– Center–Yellow	
4. Cluster– Big–Edge– Yellow		12. Ring–All– Edge–Purple		20. 8–All– Center–Green	
5. Cluster– Big–Edge– Green		13. Scratch– Long–All– Yellow		21. 8–All– Center–Purple	
6. Cluster– Big–Edge– Purple		14. Scratch– Long–All– Green		22. FlippedC– All– Center–Yellow	
7. Cluster– Big–others– Yellow		15. Scratch– Long–All– purple		23. FlippedC– All– Center–Green	
8. Cluster– Big–others– Green		16. Noise– Frequent– Scattered– Yellow		24. FlippedC– All– Center–Purple	

defining these levels. First, the number of levels should be small. Increasing the number of levels increases the complexity of the taxonomy. From the perspective of inspectors who label WBMs according to the proposed taxonomy, a simple taxonomy is easy to use with reasonable cognitive effort. Second, different frequencies among bins should be considered. As shown in Fig. 5, the frequencies of the bins were different. Frequent bins should be emphasized when defining the levels.

We clustered bins to consider these two issues. We selected the top six bins shown in Fig. 5 (bins 24, 20, 14, 28, 2, and 32) as the initial centroids. However, we found that the colors of bins 2 and 14 were similar; hence, we combined them as a single centroid. We then assigned each of the other bins to the closest centroid among the five centroids. Thus, five clusters were obtained, as shown in Fig. 9. For each cluster, we calculated the average RGB code and represented it as a color, as shown in the right part of Fig. 9. The three numbers in each square in the right part of Fig. 9 are the average RGB codes. The average RGBs of the second and third clusters were similar; hence, we combined them as a single

Table 3

Number of WBMs before and after augmentations.

Defect pattern (<i>Shape–Size–Location–Bin</i>)	Original	After 1st augmentation			After 2nd augmentation		
		Train	Validation	Test	Train	Validation	Test
1. Cluster–Big–Center–Yellow	432	302	65	65	604	130	130
2. Cluster–Big–Center–Green	101	70	15	16	621	144	144
3. Cluster–Big–Center–Purple	456	319	68	69	636	138	138
4. Cluster–Big–Edge–Yellow	5	23	5	5	621	135	135
5. Cluster–Big–Edge–Green	20	23	5	5	621	135	135
6. Cluster–Big–Edge–Purple	8	23	5	5	621	135	135
7. Cluster–Big–others–Yellow	128	89	19	20	616	140	140
8. Cluster–Big–others–Green	22	105	22	23	624	138	138
9. Cluster–Big–others–Purple	28	109	23	24	648	144	144
10. Ring–All–Edge–Yellow	19	56	12	12	616	132	132
11. Ring–All–Edge–Green	18	56	12	12	616	132	132
12. Ring–All–Edge–Purple	43	56	12	12	616	132	132
13. Scratch–Long–All–Yellow	0	63	13	14	620	140	140
14. Scratch–Long–All–Green	35	63	13	14	620	140	140
15. Scratch–Long–All–purple	55	63	13	14	620	140	140
16. Noise–Frequent–Scattered–Purple	0	9	2	2	621	138	138
17. Noise–Frequent–Scattered–Green	6	9	2	2	621	138	138
18. Noise–Frequent–Scattered–Purple	2	9	2	2	621	138	138
19. 8–All–Center–Yellow	379	265	57	57	530	114	114
20. 8–All–Center–Green	3	267	57	58	532	116	116
21. 8–All–Center–Purple	25	282	61	61	564	122	122
22. FlippedC–All–Center–Yellow	46	35	7	8	612	144	144
23. FlippedC–All–Center–Green	2	35	7	8	612	144	144
24. FlippedC–All–Center–Purple	2	35	7	8	612	144	144
Total	1835	2366	504	516	14,645	3253	3253

cluster. Thus, we obtained four clusters, as shown in the right part of [Fig. 9](#). The color of the last cluster was similar to that of bin 54 (i.e., bin of the normal chip). The four bins included in the last cluster (bins 6, 28, 37, and 48) had high spatial similarity to the bin of a normal chip, indicating that these failures were not critical. Thus, we excluded the last cluster and finally determined three levels of the bin dimension from the three clusters. They are denoted as *yellow*, *green*, and *purple* levels.

3.5. Step 5: extending SDT by adding bin dimension

We converted the probe-test data of the 4130 wafers to WBMs to visually identify the defect patterns. The 4130 wafers were selected in Step 2 to train the Bin2Vec model. Each chip was marked with a color according to its bin, as shown in [Fig. 9](#). For example, a chip with bin 24 was marked yellow. Thus, we obtained 4130 WBMs marked with four colors.

Through manual inspection of the 4130 WBMs, we identified two new defect patterns that were not defined in the SDT. [Fig. 10\(a\)](#) shows the WBM marked with white for defects, regardless of their bins. A large cluster of defects is located in the center of the WBM. In fact, this is not a simple cluster but a complex pattern composed of two bin clusters whose shape is similar to that of the number 8. However, it is impossible to identify the “8” defect pattern in [Fig. 10\(a\)](#). [Fig. 10\(b\)](#) provides more information, as the chips are marked with four colors according to their bins. Here, we observe that two clusters marked purple are located in the upper and lower areas within the large cluster marked in yellow. The shape of this new defect pattern is similar to that of an “8” or a “snowman.” Several WBMs exhibited this “8” pattern. Surprisingly, this pattern has not been previously observed, according to interviews of practitioners from the semiconductor manufacturing company. New assignable causes were identified by investigating the fabrication process, which is expected to affect the purple-colored bins. We found that both the “8” pattern and associated assignable causes resulted from new process development. Similarly, we identified another new defect pattern—a “flipped C” pattern—and associated assignable causes. These findings indicate that the proposed taxonomy can identify valid patterns.

Before extending the SDT, we modified it by adding two new patterns and excluding insignificant patterns. The sizes of the two new defect

patterns were either small or large; thus, we defined them as *8–All–Center* and *FlippedC–All–Center*. The existing SDT consists of 10 defect patterns, as shown in [Table 1](#). To exclude non-critical defect patterns, we comprehensively evaluated each defect pattern by interviewing the practitioners of the semiconductor company that provided the probe-test data. We concluded that four defect patterns with small sizes (i.e., *cluster–small–edge*, *cluster–small–others*, *noise–infrequent–scattered*, and *scratch–short–all*) were not critical; thus, we excluded them. This exclusion is consistent with that of Step 2. Thus, eight defect patterns were defined. Next, we extended each of the defect patterns using the three colors defined in Step 4. Thus, 24 (= 8 × 3) defect patterns were defined, as shown in [Table 2](#). The first 18 defect patterns in [Table 2](#) were defined by extending the existing defect patterns of the SDT, and the following six defect patterns were defined by extending the new patterns: *8–All–Center* and *FlippedC–All–Center*.

The proposed taxonomy has three advantages over existing taxonomies, including the SDT, which do not consider the bins of chips in the WBM. First, a group of major bins of the defect pattern can be identified according to the bin dimension. For example, the first three defect patterns in [Table 2](#) are considered identical in terms of the SDT; however, they are considered different in terms of the proposed taxonomy. The three patterns are likely to result from different assignable causes; thus, new assignable causes can be identified by investigating the wafers having the patterns. Second, new defect patterns can be identified. As mentioned previously, the *8–All–Center* and *FlippedC–All–Center* are new defect patterns defined in the proposed taxonomy. These patterns are likely to result from new assignable causes. Third, non-critical bins can be identified. In Step 4, we excluded the four bins included in the last cluster according to the spatial similarity to the bin of a normal chip. Because of these advantages, the proposed taxonomy can serve as a basis for various engineering activities, such as detecting and eliminating new assignable causes or analyzing the relationships between bins.

In addition, the proposed taxonomy is easy to use in practice. Typically, hundreds of bins exist in probe-test data. The bin dimension of the proposed taxonomy was developed by clustering bins according to spatial similarity, and bins in the same cluster are represented with a single color. This simplifies the taxonomy; thus, inspectors can easily use it with reasonable cognitive effort.

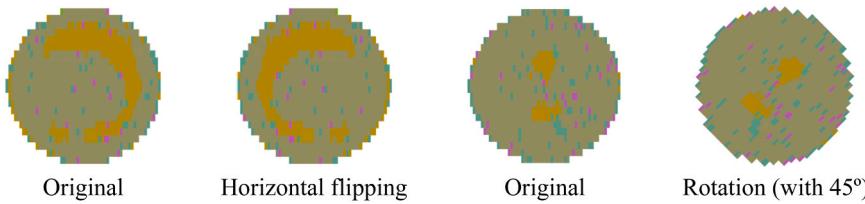


Fig. 11. Examples of augmented WBMs not defined in the proposed taxonomy.

4. Verification and validation of proposed taxonomy

To evaluate the classification performance of the proposed method, we developed a CNN classification model using the proposed taxonomy. The first task was to prepare WBMs for training the CNN model. The WBMs of 11266 wafers were collected and manually classified according to the proposed taxonomy. Each WBM was labeled as one of the defect patterns in Table 2. However, some WBMs exhibited multiple defect patterns or did not exhibit clear defect patterns that could be classified as one of the patterns in Table 1. According to (Krawczyk, 2016), wafer maps showing unclear defect patterns can negatively affect learning. Therefore, we selected 1835 WBMs with a single and clear defect pattern.

The second column of Table 3 presents the number of WBMs for each defect pattern. We found that the number of WBMs was imbalanced among the defect patterns and that some of the defect patterns had small numbers of WBMs. Such problems of small and imbalanced data are common in defect-pattern classification and should be solved to ensure high classification performance of the CNN model (Kim et al., 2021). We conducted two types of data augmentation. First, we augmented WBMs by replacing colors without modifying their shape, size, or location. The purpose of this augmentation was to resolve the imbalance of colors. For example, the 22nd pattern in Table 3 (i.e., *FlippedC-All-Center-Yellow*) had 46 WBMs, whereas the 23rd pattern (i.e., *FlippedC-All-Center-Green*) had only two WBMs. To generate the WBMs of the 23rd pattern, we replaced the yellow color of the 46 WBMs of the 22nd pattern with green. Similarly, we replaced the purple color of the two WBMs of the 24th pattern with green. The number of WBMs of the 23rd defect pattern was increased from 2 to 50 by generating 48 WBMs (= 46 + 2) via simple color replacement. Then, we divided the 50 WBMs as follows: 35 WBMs for training, 7 WBMs for validation, and 8 WBMs for testing, as shown in Table 3 (ratios of approximately 70 %, 15 %, and 15 %, respectively, which are commonly used). Thus, we augmented the WBMs of all the defect patterns and divided them into three datasets. The “after 1st augmentation” column of Table 3 presents the number of WBMs for the three datasets.

Although the 1st augmentation solved the problem of data imbalance between colors, the number of WBMs was still imbalanced among the shape, size, and location dimensions. To solve this problem, we conducted simple transformations, such as rotation, horizontal flipping, and vertical flipping, to generate augmented WBMs. We used the ImageDataGenerator library of Keras for augmentation. When conducting the transformation, we were careful not to disturb the defect patterns. For example, we did not apply horizontal flipping to “*FlippedC*” patterns, because this would result in “C,” as shown in Fig. 11. “C” patterns were not defined in the proposed taxonomy. Similarly, we did not apply rotation to “8” patterns, because the rotated “8” patterns were not defined in the proposed taxonomy. The “after 2nd augmentation” column of Table 3 presents the number of WBMs for the three datasets.

We trained the defect-pattern classification model using a CNN based on the original and augmented WBMs. Among the various CNN models, we used DenseNet201, which is the most widely used CNN structure for image classification. In addition, we used Adam as an optimizer and categorical cross-entropy as a loss function in training. We trained the CNN model with 100 epochs, and the accuracy for the validation dataset

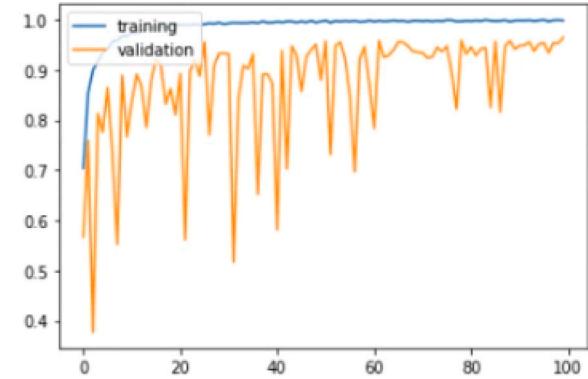


Fig. 12. Accuracy for the training and validation datasets.

Table 4
Evaluation results for the classification performance.

Defect pattern (<i>Shape-Size-Location-Bin</i>)	Precision	Recall	F1-score	Support
1. <i>Cluster-Big-Center-Yellow</i>	0.96	0.99	0.97	130
2. <i>Cluster-Big-Center-Green</i>	0.84	0.92	0.88	144
3. <i>Cluster-Big-Center-Purple</i>	0.86	0.93	0.90	138
4. <i>Cluster-Big-Edge-Yellow</i>	0.98	0.85	0.91	135
5. <i>Cluster-Big-Edge-Green</i>	0.95	0.72	0.82	135
6. <i>Cluster-Big-Edge-Purple</i>	1.00	0.85	0.92	135
7. <i>Cluster-Big-others-Yellow</i>	0.96	1.00	0.98	140
8. <i>Cluster-Big-others-Green</i>	0.67	0.96	0.96	138
9. <i>Cluster-Big-others-Purple</i>	0.90	0.97	0.94	144
10. <i>Ring-All-Edge-Yellow</i>	1.00	0.98	0.99	132
11. <i>Ring-All-Edge-Green</i>	1.00	1.00	1.00	132
12. <i>Ring-All-Edge-Purple</i>	1.00	0.99	1.00	132
13. <i>Scratch-Long-All-Yellow</i>	0.80	1.00	0.89	140
14. <i>Scratch-Long-All-Green</i>	0.98	1.00	0.99	140
15. <i>Scratch-Long-All-purple</i>	0.98	0.95	0.96	140
16. <i>Noise-Frequent-Scattered-Purple</i>	1.00	1.00	1.00	138
17. <i>Noise-Frequent-Scattered-Green</i>	0.92	1.00	0.96	138
18. <i>Noise-Frequent-Scattered-Purple</i>	0.99	1.00	1.00	138
19. 8-All-Center-Yellow	0.98	0.93	0.95	114
20. 8-All-Center-Green	0.97	1.00	0.99	116
21. 8-All-Center-Purple	0.96	0.98	0.97	122
22. <i>FlippedC-All-Center-Yellow</i>	1.00	0.88	0.93	144
23. <i>FlippedC-All-Center-Green</i>	1.00	1.00	1.00	144
24. <i>FlippedC-All-Center-Purple</i>	0.99	1.00	1.00	144
Total	0.96	0.96	0.96	

exceeded 0.95, as shown in Fig. 12.

We evaluated the classification performance of the CNN model using a test dataset consisting of 3253 WBMs. The accuracy of the CMM model was 95 % for this dataset. Additionally, we evaluated the classification performance with regard to four indices: accuracy, precision, recall, and F1-score. The results indicated that the classification performance was satisfactory, as shown in Table 4. Thus, a CNN model with high performance for defect-pattern classification can be developed using the proposed taxonomy.

In addition, we compared classification performances between the proposed taxonomy and the existing SDT to validate the proposed

Table 5

Comparison of classification performance of CNN models developed by two taxonomies.

Defect pattern	Proposed taxonomy			SDT (Kim et al., 2021)		
	Precision	Recall	F1-score	Precision	Recall	F1-score
Cluster-Big-Center	0.89	0.95	0.92	0.94	1	0.97
Cluster-Big-Edge	0.98	0.81	0.88	0.91	0.75	0.82
Cluster-Big-others	0.84	0.98	0.96	1	0.99	1
Ring-All-Edge	1.00	0.99	1.00	1	0.9	0.95
Scratch-Long-All	0.92	0.98	0.95	0.93	0.95	0.94
Noise-Frequent-Scattered	0.97	1.00	0.99	0.8	1	0.89
8-All-Center	0.97	0.97	0.97	N/A	N/A	N/A
FlippedC-All-Center	1.00	0.96	0.98	N/A	N/A	N/A
Average	0.96	0.96	0.96	0.93	0.93	0.93

taxonomy. Recently, [Kim et al. \(2021\)](#) developed the CNN model based on the SDT. The number of WBMs in the case example of [Kim et al. \(2021\)](#) was imbalanced like the case study in this section; thus, they conducted data augmentation before fitting the CNN model. Also, the CNN models of the proposed and [Kim et al. \(2021\)](#)'s method are same. Thus, we believed that the comparison is fair to investigate the effect of adding color dimension to SDT. The main difference between the proposed taxonomy and SDT is that the proposed method considers the color dimension whereas SDT does not. To resolve this issue, we merged three defect patterns having same shape-size-location into a single defect pattern. As a result, the 24 defect patterns are merged into 8 defect patterns as shown in [Table 5](#). Average of three classification performance values are calculated accordingly. The left part of [Table 5](#) reports the averages of the classification performance values. Originally, the SDT defined 10 defect patterns; however, six of them are common with the 8 defect patterns of the proposed taxonomy. Thus, classification performance indices for the common 6 defect patterns of the SDT are reported in the right part of [Table 5](#). It should be noted that the CNN model developed from the proposed taxonomy shows better classification performance compared to the CNN model from the SDT. This result indicates that considering bin information in developing taxonomy and classification model is helpful for achieving high classification performance.

5. Concluding remarks

We proposed a taxonomy composed of the shape, size, location, and bin dimension for classifying defect patterns. It extends the existing SDT by adding the bin dimension. The bin dimension was developed by employing the Bin2Vec method, which determines the RGB code for each bin according to the spatial similarity between bins. Three levels of the bin dimension were defined by analyzing a large number of WBMs using Bin2Vec and clustering methods. Compared with the existing taxonomies, the proposed taxonomy has the advantage of identifying major bins in WBMs, new defect patterns, and non-critical defect patterns. A total of 24 defect patterns were defined in the proposed taxonomy, and a CNN model for classifying the 24 patterns was developed to validate the taxonomy. A high-quality training dataset was obtained using the proposed taxonomy; thus, the CNN model exhibited satisfactory classification performance.

In this paper, the proposed method was solely applied to extend the specific taxonomy, SDT proposed by [Choi et al. \(2021\)](#). However, the proposed method is not tied to this specific taxonomy but is a universal approach that can be applied to any defect-pattern taxonomy. Any defect-pattern taxonomies be extended by adding the bin dimension through the proposed 5-step procedure. The first four steps determine RGB for each bin and form clusters based on their RGBs. Actually, the first four steps are conducted without SDT. In Step 5, SDT is extended by adding the bin dimension. In this step, SDT can be simply replaced with other taxonomy.

Recently, using CNN has been a common practice when classifying the defect patterns. One of important issues of using CNN is preparing a

high-quality training dataset to learn CNN classifier. The high-quality training dataset can be obtained when each defect pattern is labeled coherently through well-defined taxonomy. The proposed method could obtain the well-defined taxonomy by adding useful bin information. Major bins of defect patterns, new defect patterns, and non-critical defect patterns could be identified thorough the proposed method. These cannot be obtained by solely using the existing methods such as the existing taxonomies, Bin2Vec, Clustering, CNN and so on.

As a direction for future research, more systematic preprocessing and clustering are needed. In Step 2, the probe-test dataset was preprocessed through wafer screening and the preparation of bin vectors for training the Bin2Vec model. In Step 2, we selected WBMs with large or long defect patterns. For normal wafers, we selected a single wafer for each lot to represent the characteristics of the entire wafer fairly. In addition, we set $C = 12$ to generate pairs of bin vectors. In Step 4, we selected the top five bins and used them as centroids for clustering bins. We collaborated with practitioners of a semiconductor company; thus, all these decisions were based on the empirical knowledge of semiconductor process engineers. We believe that these decisions were suitable for our probe-test data; however, they may not be suitable for other probe-test data. In this regard, the development of a more systematic approach or guidelines for making optimal decisions for preprocessing and clustering is imperative.

On the other hand, the proposed method uses a neural net only to train Bin2Vec model to assign RGB code for each Bin. Once every Bin has its own RGB code, bins having similar RGB codes are clustered. This clustering is conducted on empirical judgment, but it also can be conducted with a neural net model. In order to train the neural net model for clustering, evaluation metrics such as Silhouette coefficient can be employed. Then, optimal clusters can be formed by maximizing the evaluation metric. This might reduce the amount of empirical judgment in the clustering analysis.

CRediT authorship contribution statement

Dong-Hee Lee: Methodology, Writing – original draft, Conceptualization, Formal analysis, Project administration, Writing – review & editing. **Eun-Su Kim:** Methodology, Validation, Visualization. **Seung-Hyun Choi:** Validation. **Young-Mok Bae:** Funding acquisition, Data curation. **Jong-Bum Park:** Funding acquisition, Data curation. **Young-Chan Oh:** Funding acquisition, Data curation. **Kwang-Jae Kim:** Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability

The data that has been used is confidential.

Acknowledgement

This work was supported by a research fund from SK hynix in Korea. It was also supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2022R1C1C1011743).

Disclosure statement

No potential conflict of interest was reported by the author(s).

References

- Adly, F., Alhussein, O., Yoo, P.D., Al-Hammadi, Y., Taha, K., Muhaidat, S., Jeong, Y.-S., Lee, U., Ismail, M., 2015. Simplified subspace regression network for identification of defect patterns in semiconductor wafer maps. *IEEE Trans. Ind. Inf.* 11, 1267–1276.
- Chen, F.-L., Liu, S.-F., 2000. A neural-network approach to recognize defect spatial pattern in semiconductor fabrication. *IEEE Trans. Semicond. Manuf.* 13, 366–373. <https://doi.org/10.1109/66.857947>.
- Chien, C.-F., Hsu, S.-C., Chen, Y.-J., 2013. A system for online detection and classification of wafer bin map defect patterns for manufacturing intelligence. *Int. J. Prod. Res.* 51, 2324–2338. <https://doi.org/10.1080/00207543.2012.737943>.
- Choi, G.-H., Kim, S.-H., Ha, C.-H., Bae, S.-J., 2012. Multi-step ART1 algorithm for recognition of defect patterns on semiconductor wafers. *Int. J. Prod. Res.* 50, 3274–3287. <https://doi.org/10.1080/00207543.2011.574502>.
- Choi, S.-H., Kim, E.-S., Lee, D.-H., Bae, Y.-M., Oh, Y.-C., Kim, K.-J., 2021. Development of a Spatial Dimension-based Taxonomy for Classifying the Defect Patterns of a Wafer Bin Map.
- Cunningham, S.P., MacKinnon, S., 1998. Statistical methods for visual defect metrology. *IEEE Trans. Semicond. Manuf.* 11, 48–53. <https://doi.org/10.1109/66.661284>.
- Fan, S.K.S., Lin, S.C., Tsai, P.F., 2016. Wafer fault detection and key step identification for semiconductor manufacturing using principal component analysis, AdaBoost and decision tree. *J. Ind. Prod. Eng.* 33, 151–168. <https://doi.org/10.1080/21681015.2015.1126654>.
- Hsieh, H.-W., Chen, F.-L., 2004. Recognition of defect spatial patterns in semiconductor fabrication. *Int. J. Prod. Res.* 42, 4153–4172. <https://doi.org/10.1080/00207540410001716507>.
- Hwang, J.Y., Kuo, W., 2007. Model-based clustering for integrated circuit yield enhancement. *Eur. J. Oper. Res.* 178, 143–153. <https://doi.org/10.1016/j.ejor.2005.11.032>.
- Kim, E.-S., Choi, S., Lee, D., Kim, K., Bae, Y., Oh, Y.-C., 2021. An oversampling method for wafer map defect pattern classification considering small and imbalanced data. *Comput. Ind. Eng.* 162, 107767 <https://doi.org/10.1016/j.cie.2021.107767>.
- Kim, J.-H., Kim, H.-S., Park, J.-S., Mo, K.-H., Kang, P.-S., 2019. Bin2Vec: a better wafer bin map coloring scheme for comprehensible visualization and effective bad wafer classification. *Appl. Sci.* 9, 597. <https://doi.org/10.3390/app9030597>.
- Krawczyk, B., 2016. Learning from imbalanced data: open challenges and future directions. *Prog. Artif. Intell.* 5, 221–232. <https://doi.org/10.1007/s13748-016-0094-0>.
- Kyeong, K., Kim, H., 2018. Classification of mixed-type defect patterns in wafer bin maps using convolutional neural networks. *IEEE Trans. Semicond. Manuf.* 31, 395–402. <https://doi.org/10.1109/TSM.2018.2841416>.
- Li, T.-S., Huang, C.-L., 2009. Defect spatial pattern recognition using a hybrid SOM-SVM approach in semiconductor manufacturing. *Expert Syst. Appl.* 36, 374–385. <https://doi.org/10.1016/j.eswa.2007.09.023>.
- Liu, C.W., Chien, C.F., 2013. An intelligent system for wafer bin map defect diagnosis: an empirical study for semiconductor manufacturing. *Eng. Appl. Artif. Intell.* 26, 1479–1486. <https://doi.org/10.1016/J.ENGAPPAL.2012.11.009>.
- Liukkonen, M., Hiltunen, Y., 2018. Recognition of systematic spatial patterns in silicon wafers based on SOM and K-means. *IFAC-PapersOnLine* 51, 439–444. <https://doi.org/10.1016/j.ifacol.2018.03.075>.
- Mikolov, T., Chen, K., Corrado, G., Dean, J., 2013a. Efficient estimation of word representations in vector space. In: 1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings, pp. 1–12.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J., 2013b. Distributed representations of words and phrases and their compositionality. *Adv. Neural Inf. Process Syst.* 3111–3119.
- Nakata, K., Orihara, R., Mizuoka, Y., Takagi, K., 2017. A comprehensive big-data-based monitoring system for yield enhancement in semiconductor manufacturing. *IEEE Trans. Semicond. Manuf.* 30, 339–344. <https://doi.org/10.1109/TSM.2017.2753251>.
- Nakazawa, T., Kulkarni, D.V., 2018. Wafer map defect pattern classification and image retrieval using convolutional neural network. *IEEE Trans. Semicond. Manuf.* 31, 309–314. <https://doi.org/10.1109/TSM.2018.2795466>.
- Piao, M., Jin, C.H., Lee, J.Y., Byun, J.-Y., 2018. Decision tree ensemble-based wafer map failure pattern recognition based on radon transform-based features. *IEEE Trans. Semicond. Manuf.* 31, 250–257. <https://doi.org/10.1109/TSM.2018.2806931>.
- Saqlain, M., Jargalsaikhan, B., Lee, J.Y., 2019. A voting ensemble classifier for wafer map defect patterns identification in semiconductor manufacturing. *IEEE Trans. Semicond. Manuf.* 32, 171–182. <https://doi.org/10.1109/TSM.2019.2904306>.
- Shim, J., Kang, S., Cho, S., 2020. Active learning of convolutional neural network for cost-effective wafer map pattern classification. *IEEE Trans. Semicond. Manuf.* 33, 258–266. <https://doi.org/10.1109/TSM.2020.2974867>.
- Wang, C.-H., 2009. Separation of composite defect patterns on wafer bin map using support vector clustering. *Expert Syst. Appl.* 36, 2554–2561. <https://doi.org/10.1016/j.eswa.2008.01.057>.
- Wang, C.-H., Kuo, W., Bensmail, H., 2006. Detection and classification of defect patterns on semiconductor wafers. *IIE Trans.* 38, 1059–1068. <https://doi.org/10.1080/07408170600733236>.
- Wu, M.-J., Jang, J.-S.R., Jui-Long, Chen, 2015. Wafer map failure pattern recognition and similarity ranking for large-scale data sets. *IEEE Trans. Semicond. Manuf.* 28, 1–12. <https://doi.org/10.1109/TSM.2014.2364237>.
- Yu, J., Lu, X., 2016. Wafer map defect detection and recognition using joint local and nonlocal linear discriminant analysis. *IEEE Trans. Semicond. Manuf.* 29, 33–43. <https://doi.org/10.1109/TSM.2015.2497264>.
- Yu, J., Shen, Z., Wang, S., 2021. Wafer map defect recognition based on deep transfer learning-based densely connected convolutional network and deep forest. *Eng. Appl. Artif. Intell.* 105, 104387 <https://doi.org/10.1016/J.ENGAPPAL.2021.104387>.
- Yuan, T., Kuo, W., 2007. A model-based clustering approach to the recognition of the spatial defect patterns produced during semiconductor fabrication. *IIE Trans.* 40, 93–101. <https://doi.org/10.1080/07408170701592556>.
- Yuan, T., Kuo, W., Bae, S.J., 2011. Detection of spatial defect patterns generated in semiconductor fabrication processes. *IEEE Trans. Semicond. Manuf.* 24, 392–403. <https://doi.org/10.1109/TSM.2011.2154870>.
- Zhou, Q., Zeng, L., Zhou, S., 2010. Statistical detection of defect patterns using hough transform. *IEEE Trans. Semicond. Manuf.* 23, 370–380. <https://doi.org/10.1109/TSM.2010.2048959>.