Wafer Map Failure Pattern Recognition and Similarity Ranking for Large-Scale Data Sets

Ming-Ju Wu, Jyh-Shing R. Jang, *Member, IEEE,* and Jui-Long Chen

***Abstract*—Wafer maps can exhibit specific failure patterns that provide crucial details for assisting engineers in identifying the cause of wafer pattern failures. Conventional approaches of wafer map failure pattern recognition (WMFPR) and wafer map similarity ranking (WMSR) generally involve applying raw wafer map data (i.e., without performing feature extraction). However, because increasingly more sensor data are analyzed during semiconductor fabrication, currently used approaches can be inadequate in processing large-scale data sets. Therefore, a set of novel rotation- and scale-invariant features is proposed for obtaining a reduced representation of wafer maps. Such features are crucial when employing WMFPR and WMSR to analyze large-scale data sets. To validate the performance of the proposed system, the world's largest publicly accessible data set of wafer maps was built, comprising 811 457 real-world wafer maps. The experimental results show that the proposed features and overall system can process large-scale data sets effectively and efficiently, thereby meeting the requirements of current semiconductor fabrication.**

***Index Terms*—Data models, image recognition, information retrieval, pattern recognition, semiconductor defects.**

I. Introduction

WAFER map analysis is critical in daily semiconductor manufacturing operations. Wafer maps provide visual details that are crucial for identifying the stage of manufacturing at which wafer pattern failure occurs. Experienced engineers can identify the cause of failure when a wafer map presents a specific failure pattern. However, this is a time-consuming process that requires using computer-aided tools. Furthermore, because developments in semiconductor manufacturing technology have been based on Moore's law [1] for over 50 years [2], the complexity of chip design has increased, and post analysis has become necessary to increase the yield of wafers. Concurrently, the capacity for wafer production has increased in response to the ubiquitous use of embedded and mobile devices. For instance, approximately 15 million 8-inch equivalent wafers were produced by the Taiwan Semiconductor Manufacturing Company (TSMC) in 2013 [3]. Therefore, efficient and effective wafer analysis tools are in high demand [4]–[6].

Although numerous studies have investigated wafer map failure pattern recognition (WMFPR) [7]–[17], most of them have used raw wafer maps as input data for their classification systems. However, previous approaches can be inadequate in analyzing large-scale data sets due to lower accuracy. Therefore, this paper proposes a novel set of features as a reduced representation of wafer maps. The proposed features are effective because they require minimal computation and storage, while providing discriminatory power in recognizing failure patterns, thus making them suitable for large-scale analysis of wafer maps.

This paper focuses on employing the proposed features for WMFPR and wafer map similarity ranking (WMSR). WMFPR is performed to identify wafer map failure patterns, whereas WMSR assists in retrieving similar failures in other wafer maps. To verify the performance of the proposed method, the WM-811K dataset was built comprising 811 457 wafer maps, in which each wafer map was collected from real-world fabrication. Domain experts were recruited to annotate the pattern type for approximately 20% of the wafer maps in the WM-811K dataset. The experimental results showed the feasibility of the proposed features and corresponding systems for large-scale analysis of wafer maps. In addition, TSMC has adopted the proposed system as one of their tools for wafer map analysis, thus confirming the applicability of the proposed features and systems.

The primary contributions of this paper are summarized as follows.

1) A set of features extracted from wafer maps are proposed for using WMFPR and WMSR in analyzing massive wafer maps.
2) The WM-811K dataset[1] developed in this study is the largest known wafer map data set available to the public.

The remainder of this paper is organized as follows. Section II describes the details of related work, and Section III introduces the proposed features. Sections IV and V respectively explain WMFPR and WMSR. A performance evaluation of the proposed method is presented in Section VI, and the conclusion of this study is provided in Section VII.

Manuscript received January 15, 2014; revised July 12, 2014; accepted October 14, 2014. Date of publication October 21, 2014; date of current version January 30, 2015. This work was supported by the Taiwan Semiconductor Manufacturing Company.

M.-J. Wu is with the Department of Computer Science, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: brian.wu@mirlab.org).

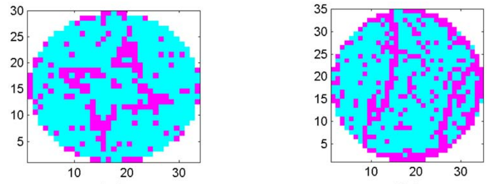
J.-S. R. Jang is with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei 10617, Taiwan (e-mail: jang@mirlab.org).

J.-L. Chen is with the Manufacturing Technology Center, Taiwan Semiconductor Manufacturing Company, Hsinchu 300-78, Taiwan (e-mail: rlchenh@tsmc.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSM.2014.2364237

[1] http://mirlab.org/dataSet/public/



(a) (b)

Fig. 1. (a) Example of a new wafer map failure pattern. Based on known failure patterns, the proposed method classified this pattern as donut, which is reasonably accurate. (b) Example of a multipattern wafer map, which is a combination of two known failure patterns (scratch and edge-ring). The proposed method classified this multipattern as scratch.

II. Related Work

Current WMFPR approaches can be divided into the following three categories:

1) *Wafer-based Clustering*
In [7] and [8], unsupervised-learning neural networks, such as adaptive resonance theory (also known as ART1), have been employed to construct clusters of wafer maps. Domain experts have labeled the clusters based on specific failure patterns, and the wafer maps have been classified based on their proximity to a cluster. An advantage of this approach is that new failure patterns can be introduced for identifying wafer maps that exhibit unknown failure patterns.

2) *Region-Based Modeling*
In [9]–[13], various shape-specific probability density functions (e.g., bivariate normal distribution, principal curve, and spherical shell) have been employed to model the regions of failure patterns. This approach is advantageous because multipattern failures can be modeled in a single wafer map.

3) *Spatial Signature Analysis*
In [14]–[16], image moments have been extracted as features from wafer maps, and a fuzzy *k*-nearest-neighbor classifier has been applied for classification. Compared with other approaches, feature extraction requires considerably less computation.

The proposed WMFPR approach can be adapted to construct new failure patterns if the confidence in classification is below a specific threshold. However, when constructing such patterns is unnecessary, the proposed method still performs acceptably in classifying input wafers to the most likely class. For instance, the class of the new failure pattern shown in Fig. 1(a) is predicted to be *Donut*, which is reasonably satisfactory. For wafers with multipattern failures, the proposed method generally predicts one of the multiple patterns. Fig. 1(b) shows a wafer map that combines two known failure patterns, *Scratch* and *Edge-ring*, which the proposed system classified as *Scratch*. Again, this result is reasonably satisfactory for classification.

Based on the WM-811K dataset and experience of domain experts, new or multipattern failures are only occasionally encountered in real-world wafer fabrication processes. Table I shows a comparison between the data sets of previous approaches and that of the proposed method. Because they lack real-world wafer data, both wafer-based clustering and region-based modeling generally depend on synthetic wafer data, which might not conform to the characteristics of actual wafer maps.

TABLE I
Comparison of Data Sets Used in WMFPR Approaches

Wafer-based clustering	14 real wafers with 35 synthetic wafers were used [7].
	240 real wafers were used [8].
Region-based modeling	Synthetic wafers were used [9].
	20 synthetic wafers were used [10], [11].
	6 real wafers with 6 synthetic wafers were used [12].
	6 real wafers with 20 synthetic wafers were used [13].
Spatial signature analysis	113 real wafers were used [14].
	1500 real wafers were used [15].
	5620 real wafers were used [16].
Proposed method	**Subset of WM-811K with 172 951 real wafers is used.**

TABLE II
Comparison of WMFPR Approaches

	Rotation-invariant	Feature-based analysis	For large-scale data sets
Wafer-based clustering			
Region-based modeling	●		
Spatial signature analysis	●	●	
Proposed method	●	●	●

The comparison of WMFPR approaches in Table II shows that *wafer-based clustering* does not preserve the rotation-invariant attribute [17]. In other words, two wafers with an identical failure pattern but distinct rotation degrees may be considered different failure patterns. In addition, both *wafer-based clustering* and *region-based modeling* apply the raw wafer maps rather than the extracted features as the input for further processing. Although *spatial signature analysis* extracts features from wafer maps, the involved fuzzy *k*-nearest-neighbor classifier remains computationally expensive when classifying large-scale data sets. Conversely, the proposed WMFPR applies support vector machines (SVMs) as the classifier, which is highly efficient because the failure pattern prediction is determined by only a few critical training instances (i.e., the support vectors). By combining the proposed features and SVM classifier, the proposed WMFPR can predict failure patterns with acceptable accuracy and high throughput.

Research on WMSRs is scant. In [18], wafer maps were input into an SVM classifier to determine the degree of similarity. Again, using raw wafer maps as the input is inefficient for searching large-scale data sets. By contrast, because the proposed WMSR is based on the proposed features, similar wafer maps can be retrieved efficiently on a large scale.

III. Feature Extraction

Effective features are vital for the success of pattern recognition applications. In this study, discriminative features were

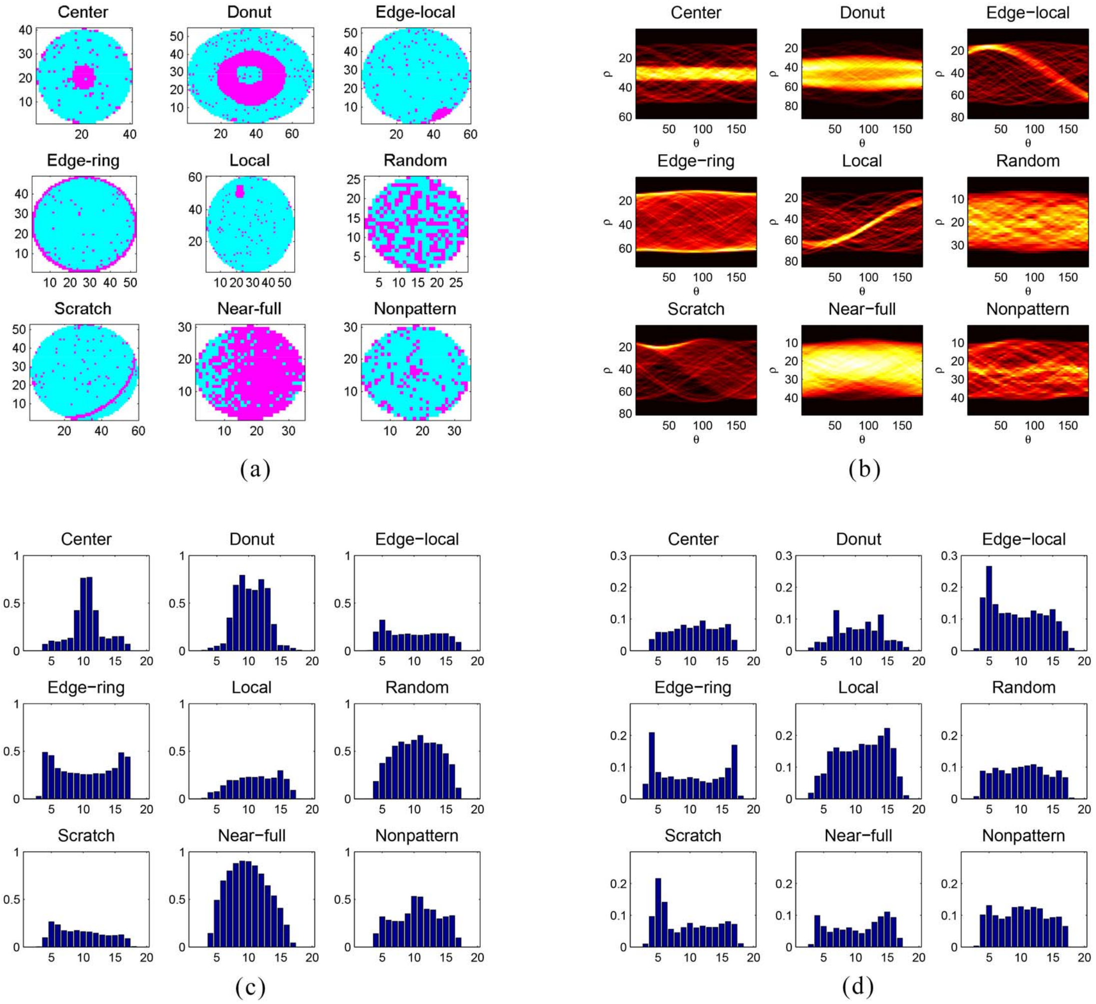


Fig. 2. (a) Typical examples of wafer map failure types. (b) Projection results **G** following the Radon transform. (c) Radon-based features $\mathbf{R}_{\mu}$. (d) Radon-based features $\mathbf{R}_{\sigma}$.

extracted from each wafer map to form a reduced representation for subsequent classification and analysis. This section introduces the proposed Radon- and geometry-based features. The Radon-based features are based on the projection of wafer maps along various directions, whereas the geometry-based features are based on the geometric measures of regions obtained from wafer maps. Subsequently, these two types of feature are concatenated to form a new representation of each wafer map.

A. Radon-Based Features

The proposed Radon-based features are based on the Radon transform [19], which has been used successfully in computed tomography for medical applications. The Radon transform can generate a 2D representation of a wafer map according to a series of projections. The concept of the Radon transform is detailed as follows.

First, (1) is used to represent a line

$$x\cos\theta + y\sin\theta = \rho \tag{1}$$

where ρ denotes the distance between the line and the point of origin, and θ denotes the angle from the x-axis. Here, the projection is performed along each straight line with specific values of ρ and θ. Thus, the Radon transform can be expressed as

$$g(\rho,\theta) = \sum_{x=1}^{m}\sum_{y=1}^{n}\mathbf{M}(x,y)\delta(x\cos\theta + y\sin\theta - \rho) \tag{2}$$

where **M** is a wafer map of size $m \times n$. Each element in **M** is set at 1 to indicate a defective die, and 0 otherwise. $g(\rho,\theta)$ is the response of a projection, and δ is the impulse function

$$\delta(k) = \begin{cases} 1, \text{ if } k = 0 \\ 0, \text{ otherwise} \end{cases} \tag{3}$$

By varying ρ and θ, the response of the Radon transform can be expressed as

$$\mathbf{G} = \begin{pmatrix} g(1,1) & g(1,2) & \cdots & g(1,\theta_{max}) \\ g(2,1) & g(2,2) & \cdots & g(2,\theta_{max}) \\ \vdots & \vdots & \ddots & \vdots \\ g(\rho_{max},1) & g(\rho_{max},2) & \cdots & g\,(\rho_{max},,\theta_{max}) \end{pmatrix} \tag{4}$$

To ensure that response **G** is comparable among wafer maps, minmax normalization is applied to **G**

$$\mathbf{G}' = \frac{\mathbf{G} - \min(\mathbf{G})}{\max(\mathbf{G}) - \min(\mathbf{G})} \tag{5}$$

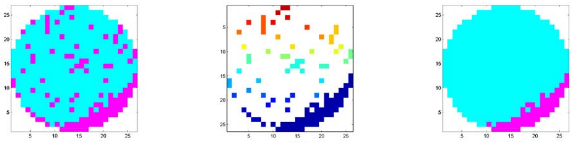

(a) (b) (c)

Fig. 3. Process for identifying the most salient region of a wafer map. (a) Original wafer map. (b) Results of the region-labeling algorithm, where each region is assigned a different color. (c) Most salient region with the maximal area (in this example, it is also the region with the maximal perimeter).

(For simplicity, **G** represents the response following normalization).

Fig. 2(a) shows several typical examples of wafer map failure types (with the exception of *Nonpattern*, which indicates no specific failure pattern). Fig. 2(b) shows the corresponding result of the Radon transform (**G**), where the x- and y-axes represent θ and ρ, respectively. The figure shows that the 2D representations obtained from the Radon transform effectively represent the structure of the failure patterns. For example, *Center* has a narrow but strong response in the central area of ρ. *Donut* is similar to *Center*, except the response in the central area is wider. By contrast, *Edge-ring* exhibits a strong response along the border of the response band.

To extract the Radon-based features, row mean $\mathbf{G}_\mu$ and row standard deviation $\mathbf{G}_\sigma$ are calculated from **G**, where $\mathbf{G}_\mu$ is the mean response of the Radon transform over θ, and $\mathbf{G}_\sigma$ is the variance of the response of the Radon transform over θ. Next, $\mathbf{G}_\mu$ and $\mathbf{G}_\sigma$ are respectively resampled using cubic interpolation [20] to obtain the Radon-based features $\mathbf{R}_\mu$ and $\mathbf{R}_\sigma$, each of which has experimentally fixed dimensions of 20. The $\mathbf{R}_\mu$ and $\mathbf{R}_\sigma$ appear to be rotation-invariant and scale-invariant. Fig. 2(c) and (d) respectively show $\mathbf{R}_\mu$ and $\mathbf{R}_\sigma$ of the corresponding nine wafer maps, demonstrating that the Radon-based features exhibit discriminatory power among the various types of failure.

B. Geometry-Based Features

Geometry-based features are used to measure the geometric attributes of each wafer map. Based on observations of numerous wafer maps and consultations with domain experts, the geometry-based features were obtained by calculating the regional, statistical, and linear attributes, all of which are rotation- and scale-invariant and are detailed as follows.

1) Regional Attributes: The connected defective dice in a wafer map form regions that can indicate specific failure patterns. Because wafer maps can exhibit multiple regions, the most salient region (i.e., the maximal area or perimeter of a wafer map) was examined (Fig. 3). Fig. 3(a) shows an original wafer map. Fig. 3(b) shows the results of the region-labeling algorithm [21] (yielding 26 regions), each of which was assigned a different color. Finally, Fig. 3(c) shows the most salient region with the maximal area.

Table III lists the attributes for the most salient region, as determined according to the experimental results. Let s_a and s_p indicate the most salient region with the maximal area and the maximal perimeter, respectively. Because the wafer maps vary in size, the attributes must be normalized by dividing appropriate constants, as shown in (6)–(11), detailed as follows:

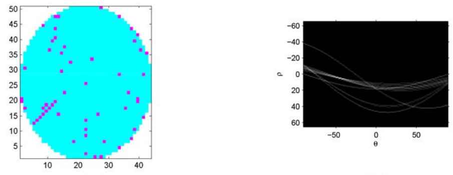
TABLE III
Regional Attributes

$\dfrac{a}{\pi r^2}$	(6)
$\dfrac{p}{r}$	(7)
$\max\limits_{(x,y)\in S_a} \dfrac{\sqrt{(x-c_x)^2+(y-c_y)^2}}{r}$, $\max\limits_{(x,y)\in S_p} \dfrac{\sqrt{(x-c_x)^2+(y-c_y)^2}}{r}$	(8)
$\min\limits_{(x,y)\in S_a} \dfrac{\sqrt{(x-c_x)^2+(y-c_y)^2}}{r}$, $\min\limits_{(x,y)\in S_p} \dfrac{\sqrt{(x-c_x)^2+(y-c_y)^2}}{r}$	(9)
$\dfrac{m_j}{r}$	(10)
$\dfrac{m_i}{r}$	(11)
$\dfrac{a}{a_{convex}}$	(12)
$\sqrt{1-\dfrac{{m_i}^2}{{m_j}^2}}$	(13)

[1] a= area of a region; p= perimeter of a region; r= radius of a wafer map; (x, y)= position of a defective die within a region; (c_x, c_y)= center of a wafer map; m_j= length of the major axes of the estimated eclipses surrounding a region; m_i= length of the minor axes of the estimated eclipses surrounding a region; a_{convex}= area of the smallest convex polygon that can contain a region.

- (6) Ratio of the area of s_a to the area of the wafer map.
- (7) Ratio of the perimeter of s_p to the radius of the wafer map.
- (8) Maximal distance between s_a (or s_p) and the center of the wafer map.
- (9) Minimal distance between s_a (or s_p) and the center of the wafer map.
- (10) Ratio of the length of the major axes of the estimated eclipses surrounding s_a (or s_p).
- (11) Ratio of the length of the minor axes of the estimated eclipses surrounding s_a (or s_p).
- (12) Solidity, indicating the proportion of defective dice in the estimated convex hull in s_a (or s_p).
- (13) Eccentricity, indicating the shape of the estimated eclipse surrounding s_a (or s_p), where the value is 0 for a circle, or 1 for a line.

2) Statistical Attributes: The number of defective dice (NDD) is a useful statistic for wafer maps [22]. Because



(a) (b)

Fig. 4. Example of line detection by using the Hough transform. (a) Scratch wafer map. (b) Result of the Hough transform for (a). Accumulator **A** has local maxima at approximately $(\rho, \theta) = (-1, -45)$, which indicates the line segment in (a).

wafer maps vary in size, the ratio of defective dice (RDD) was measured for each wafer map to indicate the degree of failure. In particular, the global and local RDDs were measured, where the global RDD represents the entire wafer map, and the local RDD represents specific zones of a wafer map. In this study, the local RDD was measured for the two outer-most rings (based on an 8-connected neighborhood) of each wafer map, because defective dice that occur at the boundary of a wafer map tend to indicate a specific type of failure, such as *Edge-ring* and *Edge-local*.

3) Linear Attribute: The Hough transform [19], which was applied in [23], was used in the current study to detect the lines in each wafer map. The Hough transform algorithm is detailed as follows. First, the 2D accumulator **A** must be created to indicate the possibility of lines occurring in a map. For each defective die in a wafer map, the following two steps must be performed to obtain **A**

1) *Given a defective die's position* (x, y)*, find a set* $\{(\rho, \theta)\,|x\cos\theta + y\sin\theta = \rho\}$*. Each pair of* (ρ, θ) *represents the corresponding virtual line passing through* (x, y)*.*
2) *Increase* $\mathbf{A}(\rho, \theta)$ *by 1 for each instance of* (ρ, θ)*.*

After all defective dice in a wafer map are considered, the local maxima in **A** indicates the possible lines within that map. Fig. 4 shows an example of the Hough transform. Fig. 4(a) shows a *Scratch* wafer map, and Fig. 4(b) depicts the result of the Hough transform. Here, θ is measured clockwise relative to the positive x-axis. The local maxima occur at $(\rho, \theta) = (-1, -45)$, indicating the line segment in the original wafer map. In this study, a line is established when its length is longer the 10% of the wafer's diameter. Moreover, a broken line is identified as a line when the length of a gap is less than 3% of the wafer's diameter. Finally, the number of detected lines is used as the feature.

IV. Wafer Map Failure Pattern Recognition

This section introduces the WMFPR, which involves using the proposed features. The flowchart in Fig. 5 shows that the WMFPR is based on a two-stage framework. Stage 1 entails determining whether a wafer map exhibits a failure pattern, and Stage 2 involves identifying the pattern type. An SVM is used as a classifier at each stage because of its superior efficiency in large-scale data set applications [24]. During the training phase, the SVMs determine the support vectors in the training data, which are applied to predict new wafer maps during the test phase. The main advantage of the two-stage framework is that the parameters can be tuned to optimize the tradeoff between the false-positive rate and the false-negative rate at Stage 1. The basic concept of the SVM is described in Section IV-A, and the maintenance of the ground truth is explained in Section IV-B.

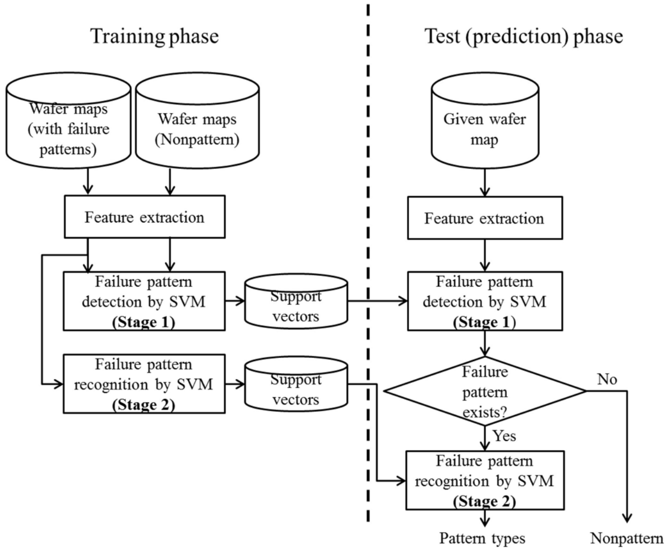

Training phase
Test (prediction) phase
Wafer maps (with failure patterns)
Wafer maps (Nonpattern)
Given wafer map
Feature extraction
Feature extraction
Failure pattern detection by SVM (Stage 1)
Support vectors
Failure pattern detection by SVM (Stage 1)
Failure pattern recognition by SVM (Stage 2)
Support vectors
Failure pattern exists?
No
Yes
Failure pattern recognition by SVM (Stage 2)
Pattern types
Nonpattern

Fig. 5. Flowchart of the proposed WMFPR. Stage 1: the SVM determines whether a failure pattern exists. Stage 2: the SVM identifies the wafer map failure pattern.

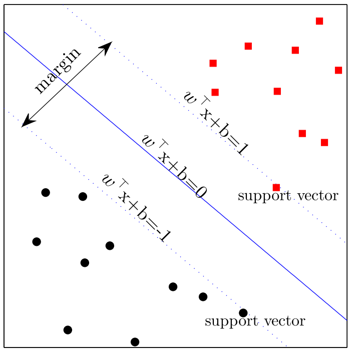


Fig. 6. Example of the hyperplane (i.e., decision boundary) of an SVM. An SVM is designed to determine the maximum-margin hyperplane separating two classes of data.

A. Support Vector Machine for the WMFPR

The function of an SVM [25] is to identify the hyperplane (i.e., decision boundary) with the widest separation between two classes of training data, as shown in Fig. 6. The hyperplane in Fig. 6 is expressed as $\mathbf{w}^T\mathbf{x} + b = 0$, which satisfies the constraints in (14)

$$\begin{cases} \mathbf{w}^T\mathbf{x}_i + b \geq 1 & \forall y_i \in 1 \\ \mathbf{w}^T\mathbf{x}_i + b \leq -1 & \forall y_i \in -1 \end{cases} \tag{14}$$

where $\mathbf{w}$ is a normal vector, b is the bias term in the hyperplane, $\mathbf{x}_i$ is a d-dimensional feature vector of a wafer map, and y_i is the label of $\mathbf{x}_i$, which is set at either 1 or -1 to distinguish between the two classes. The training set can be expressed as

$\{(\mathbf{x}_i, y_i)|\mathbf{x}_i \in R^d, y_i \in \{1, -1\}, \forall i = 1, \ldots, l\}$, where l denotes the number of wafer maps in the training set.

An SVM is designed to determine the hyperplane at which the margin between two classes of data is maximized, where the margin is the distance between $\mathbf{w}^T x_i + b = \pm 1$. A wider margin indicates that the classifier exhibits superior generalization capability. However, the constraint of (14) is too strict when an SVM is applied to nonseparable data sets, resulting in the nonexistence of such a hyperplane. This problem is solved by introducing the slack variable ξ_i, which allows not every $\mathbf{x}_i$ to be necessary on the right side of the hyperplane. Then (14) can be reformulated as

$$y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) \geq 1 - \xi_i \tag{15}$$

where ξ_i applies to one of the following three cases

$$\begin{cases} \text{If } \mathbf{x}_i \text{ is correctly classified and outside the margin,} \\ \quad \text{then } \xi_i = 0. \\ \text{If } \mathbf{x}_i \text{ is correctly classified and inside the margin,} \\ \quad \text{then } 0 < \xi_i \leq 1. \\ \text{If } \mathbf{x}_i \text{ is missclassified,} \\ \quad \text{then } \xi_i > 1. \end{cases} \tag{16}$$

Next, the search for the optimal hyperplane can be formulated as the following constrained optimization problem

$$\begin{cases} \min J(\mathbf{w}, b, \xi) = \frac{1}{2}\|\mathbf{w}\|^2 + c\sum\limits_{i=1}^{l} \xi_i \\ \text{subject to } y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) \geq 1 - \xi_i, \\ \qquad \xi_i \geq 0, \forall i = 1, \ldots, l \end{cases} \tag{17}$$

where c is a predefined cost value. Because (17) involves an inequality constraint, the Lagrange multiplier and Karush-Kuhn-Tucker conditions are applied to solve the problem. Introducing the Lagrange multiplier enables (17) to be reformulated as follows

$$\begin{cases} \min L(\mathbf{w}, b, \xi, \alpha, \lambda) = \frac{1}{2}\|\mathbf{w}\|^2 + c\sum\limits_{i=1}^{l} \xi_i - \sum\limits_{i=1}^{l} \alpha_i \xi_i \\ \qquad - \sum\limits_{i=1}^{l} \lambda_i \left[y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) - 1 + \xi_i\right] \\ \text{subject to } \alpha_i, \lambda_i \geq 0, \forall i = 1, \ldots, l \end{cases} \tag{18}$$

The corresponding Karush-Kuhn-Tucker conditions are

$$\begin{cases} \frac{\partial L}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum\limits_{i=1}^{l} \lambda_i y_i \mathbf{x}_i \\ \frac{\partial L}{\partial b} = 0 \Rightarrow \sum\limits_{i=1}^{l} \lambda_i y_i = 0 \end{cases} \tag{19}$$

The hyperplane can then be expressed as

$$g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b = \sum_{i=1}^{l} \lambda_i y_i \mathbf{x}_i^T \mathbf{x} + b = \sum_{i=1}^{N_s} \lambda_i y_i \mathbf{x}_i^T \mathbf{x} + b \tag{20}$$

Notably, the Lagrange multiplier λ_i can be either zero or positive. In other words, the optimal hyperplane is the linear combination of $\mathbf{x}_i$ with $\lambda_i > 0$; these $\mathbf{x}_i$ are called support vectors, which support the maximum-margin and create the optimal hyperplane.

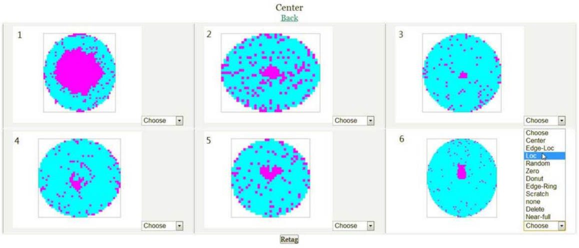


Fig. 7. Examples of the support vectors for the failure type center based on the screenshot of the web interface. Domain experts must inspect the support vectors further to ensure that the ground truth labeling is accurate for two reasons. First, support vectors tend to be close to the maximum-margin hyperplane; consequently, they are more likely to be mislabeled. Second, SVM prediction is determined only by support vectors; labeling them inappropriately is detrimental to the accuracy of an SVM.

Moreover, a linear mapping ϕ can be applied to transform each $\mathbf{x}_i$ into a new space with high dimensionality to facilitate data separation. According to the *kernel trick*, the inner product of the new space has the equivalent representation in the original space

$$\left\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \right\rangle = K\left(\mathbf{x}_i, \mathbf{x}_j\right) \tag{21}$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ are feature vectors. This indicates that the inner product of $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$ can be obtained without explicitly computing $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$. Consequently, the optimal hyperplane also has a similar representation

$$g(\mathbf{x}) = \sum_{i=1}^{N_s} \lambda_i y_i \phi(\mathbf{x}_i)^T \phi(\mathbf{x}) + b = \sum_{i=1}^{N_s} \lambda_i y_i K(\mathbf{x}_i, \mathbf{x}) + b \tag{22}$$

where N_s denotes the number of support vectors. Here, the widely used Radial basis function (RBF) kernel [26] is adopted in (23)

$$K(\mathbf{x}_i, \mathbf{x}) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}\|^2}{\sigma^2}\right) \tag{23}$$

Equations (22) and (23) can then be used to predict the class of feature vector $\mathbf{x}$ of a new wafer, depending on whether the sign of $g(\mathbf{x})$ is positive or negative. However, the mentioned SVM is used for binary-class classification, which is applicable at Stage 1. At Stage 2, eight pattern types require classification, which can be achieved by applying the *one-against-one* technique. Consequently, 8(8-1)/2= 28 SVM classifiers can be constructed for each pair of classes. Then the pattern type for a wafer map is predicted based on the maximal vote among the 28 classifiers.

B. Ground Truth Maintenance With Support Vectors

Because support vectors are used for supporting the maximal margin, they tend to be close to the maximum-margin hyperplane; consequently, support vectors are more likely to be mislabeled. In general, if these support vectors are labeled appropriately, they are likely to be located at the boundary of the feature space of a specific class. Fig. 7 shows typical examples of the support vectors of *Center*, in which the appearance of these support vectors are diverse, and some of which could be ambiguous. For example, it would not be unreasonable to label Wafer Map 4 as *Donut*, whereas Wafer

Maps 5 and 6 could be labeled as *Local*. Because SVM prediction is determined using support vectors alone, labeling them inappropriately is detrimental to the accuracy of an SVM. Therefore, we provided a web interface for domain experts who inspected and relabeled the support vectors, and the SVM was retrained to improve the robustness of the recognition system. In particular, this inspection process can effectively stabilize the recognition system when new failure patterns are introduced.

V. Wafer Map Similarity Ranking

WMSR is used to retrieve wafer maps that are similar to a given queried wafer map. Because wafer maps with similar failure patterns tend to have identical failure causes, WMSR can assists engineers in identifying the root cause of similar failure patterns. The framework of the proposed WMSR involves two stages:

- Stage 1: Search the top-*n* similar wafer maps based on the Euclidean distance of the extracted features.
- Stage 2: Rank the top-*n* candidates (from Stage 1) based on the 2D normalized correlation coefficient (i.e., known as template matching [19] in the field of image processing).

Because wafer maps vary in size, it is necessary to first normalize the size of both the queried wafer map and all wafer maps in the dataset. Then 2D normalized correlation coefficient in (24) can be computed to obtain the similarity score between two wafer maps.

$$s(\mathbf{Q},\mathbf{C}) = \sum_{x=1}^{m}\sum_{y=1}^{n}\left[\mathbf{Q}(x,y)-\overline{\mathbf{Q}}\right]\left[\mathbf{C}(x,y)-\overline{\mathbf{C}}\right] \cdot \left(\sum_{x=1}^{m}\sum_{y=1}^{n}\left[\mathbf{Q}(x,y)-\overline{\mathbf{Q}}\right]^2 \sum_{x=1}^{m}\sum_{y=1}^{n}\left[\mathbf{C}(x,y)-\overline{\mathbf{C}}\right]^2\right)^{-0.5} \tag{24}$$

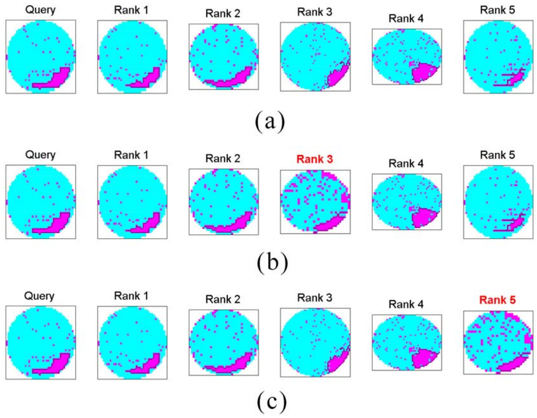
where $\mathbf{Q}$ denotes a queried wafer map with size $m \times n$, $\overline{\mathbf{Q}}$ is the mean of $\mathbf{Q}$, $\mathbf{C}$ represents a candidate wafer map, and $\overline{\mathbf{C}}$ is the mean of $\mathbf{C}$. The similarity score s($\mathbf{Q}$, $\mathbf{C}$) ranges from -1 to 1, where a higher value indicates greater similarity.

In addition, the similarity between the most salient regions should also be considered (Fig. 3(c) shows an example of the most salient region). Therefore, the similarity score for Stage 2 is expressed as

$$similarity(\mathbf{Q},\mathbf{C}) = \beta s(\mathbf{Q},\mathbf{C}) + (1-\beta)s\left(\mathbf{Q}',\mathbf{C}'\right), \quad 0 \le \beta \le 1 \tag{25}$$

where s($\mathbf{Q}'$, $\mathbf{C}'$) is the score between the most salient regions of a query and candidate wafer maps. β represents the weighting of s($\mathbf{Q}$, $\mathbf{C}$) and s($\mathbf{Q}'$, $\mathbf{C}'$). Finally, after the similarity scores in (25) are computed for all of the candidate wafer maps, the similarity scores are sorted in descending order to obtain the similarity ranking.

Fig. 8 shows the results of three examples of WMSR when β was respectively set to 1, 0, and 0.5 in Fig. 8(a)–(c), respectively. The leftmost wafer map is the input query wafer map, whereas the other wafer maps were retrieved from the training set (the leftmost column lists three identical input query wafer maps). For each wafer map, the boundary of the most salient region was surrounded by a black outline. As shown in Fig. 8(a), the results were based on only $s(\mathbf{Q},\mathbf{C})$. As shown in Fig. 8(b), the results were based on only $s(\mathbf{Q}',\mathbf{C}')$, leading to the retrieval of wafer maps with similar regions but dissimilar noise, such as the rank-3 wafer map. Fig. 8(c) shows the results were the fusion of $s(\mathbf{Q},\mathbf{C})$ and $s(\mathbf{Q}',\mathbf{C}')$, which typically demonstrate highly desirable performance. Specifically, the wafer map that was ranked 5 in Fig. 8(c) is typically a more favorable candidate than the wafer map that was ranked 5 in Fig. 8(a) and (b). Therefore, β was set to 0.5 in this study.



(a)

(b)

(c)

Fig. 8. The top-five similar wafer maps were obtained from the training set by using different β in (25). The boundary of the most salient region of each wafer map was surrounded by a black outline. (a) $\beta = 1$, indicating the similarity score is based on $s(\mathbf{Q},\mathbf{C})$ only. (b) $\beta = 0$, indicating the similarity score is based on $s(\mathbf{Q}',\mathbf{C}')$ only. (c) $\beta = 0.5$, indicating the similarity score is the average of $s(\mathbf{Q},\mathbf{C})$ and $s(\mathbf{Q}',\mathbf{C}')$.

VI. Performance Evaluation

This section introduces the WM-811K dataset (Section VI-A) and experimental settings (Section VI-B), and presents the results of using the proposed WMFPR (Section VI-C) and WMSR (Section VI-D), as well as their computation times (Section VI-E).

A. Data Collection

The WM-811K dataset comprises 811 457 wafer maps that were collected from 46 293 lots in real-world fabrication. Although each lot should contain 25 wafer maps, some were blank (and thus removed) because of sensor failure or for other unknown reasons. The histogram of the number of dice in Fig. 9 shows that the number of dice varies considerably. Specifically, there are only 696 599 unique wafer maps, regardless of their failure bin types (the element of a wafer map is set at 1 to indicate good dice or 2 to indicate defective dice).

The data set was divided into a training set (to construct the recognition system) and a test set (to test the system performance). For creating the training set, a diverse range of wafer maps were selected to include each pattern type to ensure that the constructed model would be robust. Conversely, the test set comprised wafer maps that were randomly selected by domain experts. Approximately 20% of the wafer maps

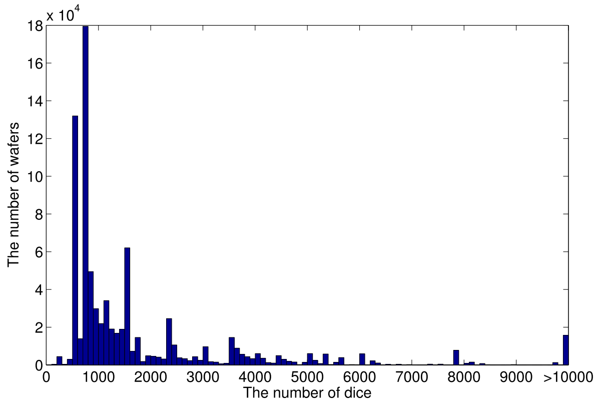
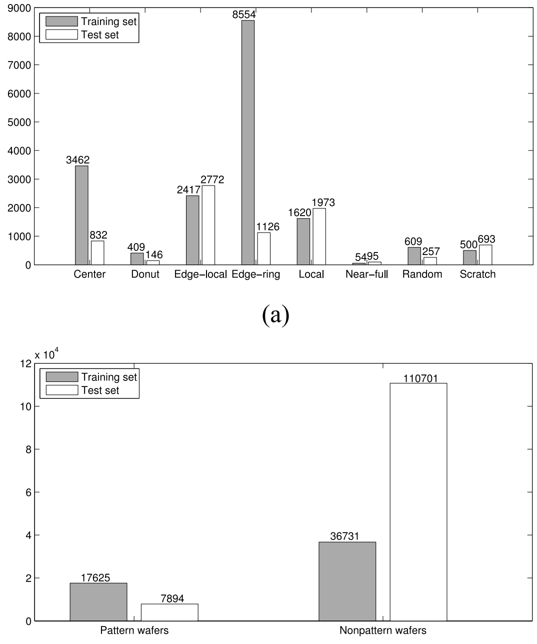


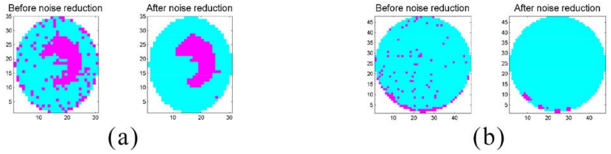
Fig. 9. Histogram of the number of dice for the wafer maps in the WM-811K data set.



(a)

(b)

Fig. 10. Wafer map distribution in training and test sets. (a) Distribution of wafer map failure patterns (eight types). (b) Distribution of pattern and nonpattern wafer maps.



(a)

(b)

Fig. 11. Effects of noise reduction. (a) Noise reduction successfully exposes donut. (b) Same noise reduction destructs edge-ring.

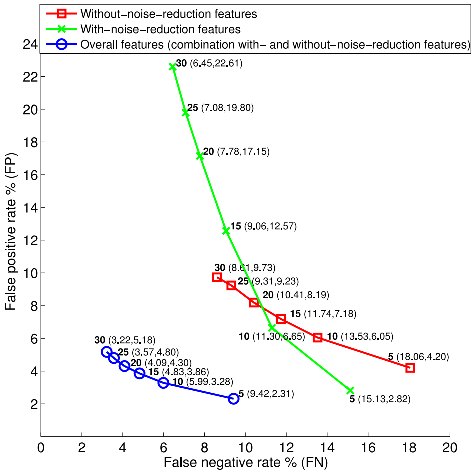


Fig. 12. Detection error tradeoff (DET) curve for various feature sets for the WMFPR, where the value of $c_{\text{nonpattern}}$ was fixed at 1 to determine an optimum value of c_{pattern} in the SVM. The bold annotation represents the value of c_{pattern}. The overall features achieved the optimal curve in the DET plot.

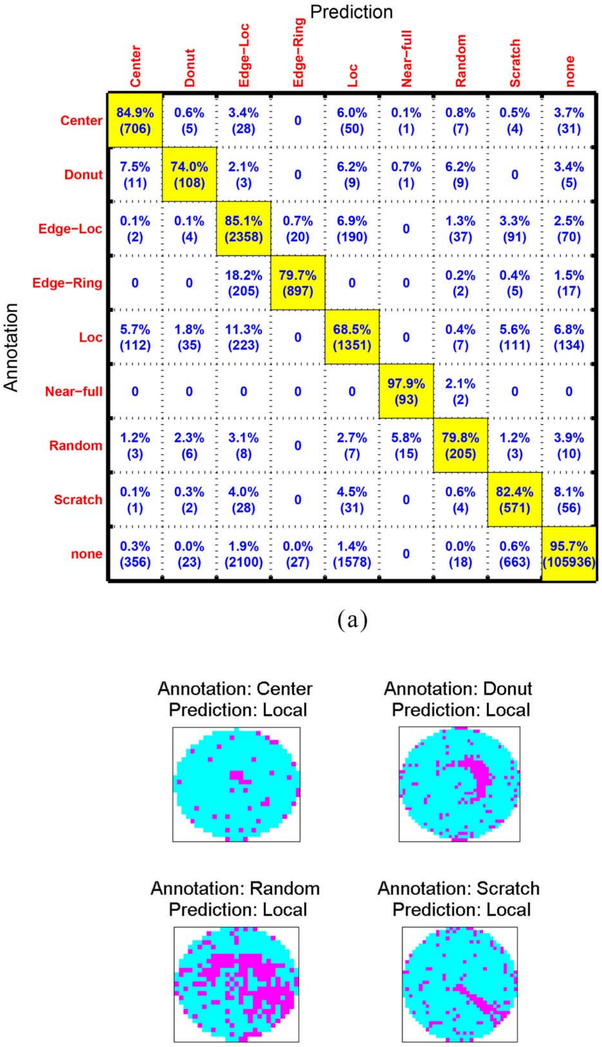
were labeled from one of the nine types (54 356 in the training set and 118 595 in the test set); each type is shown in Fig. 2(a), which includes *Center*, *Donut*, *Edge-local*, *Edge-ring*, *Local*, *Near-full*, *Random*, *Scratch* and *Nonpattern* (the first eight types are regarded as *Pattern*). In addition, both training and test sets comprised unique wafer maps. Fig. 10 shows the distributions for both sets.

B. Experimental Settings

Wafer maps are generally accompanied by noise. However, identifying a single noise reduction scheme that can reduce noise only without producing detrimental effects on crucial patterns is difficult. As shown in Fig. 11, noise reduction effectively exposes *Donut* in Fig. 11(a). Conversely, the same noise reduction destructs *Edge-ring* in Fig. 11(b). In this study, feature extraction was independently performed with and without noise reduction for each wafer map. The combination with- and without- noise-reduction features was used as the overall features, because it was verified to achieve optimal performance (Fig. 12). The dimensions of the geometry-based and Radon-based features were 18 and 40, respectively. Consequently, the overall feature dimension was $(18 + 40) \times 2 = 116$. The noise reduction was based on the median filter. For the classifier, the well-known SVM tool LIBSVM [27] was adopted. The OpenMP library was used to enable the LIBSVM to support the parallel computing technique. The evaluation environment was operated on a personal computer with an Intel Core i7 2600 CPU (4 cores), 16 GB RAM, and MATLAB R2012a.

C. Failure Pattern Recognition Results

In using the proposed WMFPR, Stage 1 was designed to identify whether each wafer map exhibited a failure pattern. Here, false-positive (FP) is defined as the rate of misclassifying *Nonpattern* as *Pattern*, whereas false-negative (FN) is the rate of misclassifying *Pattern* as *Nonpattern*. The tradeoff between FP and FN can be adjusted using c in (17). Let $c_{nonpattern}$ and $c_{pattern}$ denote the cost value of all *Nonpattern* and *Pattern* wafer maps in the training set, respectively. The value of $c_{nonpattern}$ was fixed at 1 for determining an appropriate value of $c_{pattern}$ to obtain reasonable results for both FP and FN. Fig. 12 shows the detection error tradeoff (DET) curve obtained using various feature sets when the value of $c_{pattern}$ varies. Specifically, an SVM was trained according to



	Center	Donut	Edge-Loc	Edge-Ring	Loc	Near-full	Random	Scratch	none
Center	84.9% (706)	0.6% (5)	3.4% (28)	0	6.0% (50)	0.1% (1)	0.8% (7)	0.5% (4)	3.7% (31)
Donut	7.5% (11)	74.0% (108)	2.1% (3)	0	6.2% (9)	0.7% (1)	6.2% (9)	0	3.4% (5)
Edge-Loc	0.1% (2)	0.1% (4)	85.1% (2358)	0.7% (20)	6.9% (190)	0	1.3% (37)	3.3% (91)	2.5% (70)
Edge-Ring	0	0	18.2% (205)	79.7% (897)	0	0	0.2% (2)	0.4% (5)	1.5% (17)
Loc	5.7% (112)	1.8% (35)	11.3% (223)	0	68.5% (1351)	0	0.4% (7)	5.6% (111)	6.8% (134)
Near-full	0	0	0	0	0	97.9% (93)	2.1% (2)	0	0
Random	1.2% (3)	2.3% (6)	3.1% (8)	0	2.7% (7)	5.8% (15)	79.8% (205)	1.2% (3)	3.9% (10)
Scratch	0.1% (1)	0.3% (2)	4.0% (28)	0	4.5% (31)	0	0.6% (4)	82.4% (571)	8.1% (56)
none	0.3% (356)	0.0% (23)	1.9% (2100)	0.0% (27)	1.4% (1578)	0	0.0% (18)	0.6% (663)	95.7% (105936)

(a)

(b)

Fig. 13. (a) Combined confusion matrix for Stages 1 and 2 on the test set. (b) Wafer maps are easily confused with local, although users generally accept the predictions because these wafer maps seem to saddle across the boundary of two types.

each value of and the resulting FN and FP were reported in the test set. In this figure, the bold annotation indicates the values of $c_{pattern}$. In general, a DET curve close to the origin indicates favorable performance. FN was less sensitive to $c_{pattern}$ when with-noise-reduction features were used. Conversely, FP was less sensitive to $c_{pattern}$ when without-noise-reduction features were used. This indicates the two feature sets may contain distinct discriminative information. As expected, the overall features (the combination with- and without-noise-reduction features) can be used to achieve the optimal curve in the DET plot. When $c_{pattern}$ is set at 20 (the closet point to the origin) for evaluating the overall features, minimal error is produced. Therefore, the overall features with $c_{pattern}$= 20 were applied in this study.

TABLE IV
ACCURACY COMPARISON FOR WMFPR

Approach	Accuracy
Proposed method	94.63%
Deep learning	89.64%
Spatial signature analysis	76.46%

At Stage 2, the proposed WMFPR identified the wafer map failure pattern after it was classified as a *Pattern* wafer. Fig. 10 shows that the number of samples for each failure type in the WM-811K dataset was unevenly distributed. However, each failure pattern type is equally crucial, regardless of the number of samples for a specific failure pattern. Therefore, a relatively higher cost value c was assigned to the pattern types with relatively fewer samples. Specifically, the new cost value for each pattern type was proportional to the inverse of its corresponding sample size in the training set.

Fig. 13(a) shows the combined confusion matrix for Stages 1 and 2 on the test set (overall accuracy = 94.63%). In the figure, the annotations (ground truth) are shown in the left column, and the predictions by the proposed system are in the top row. The diagonal elements represent the recognition rate of each type. The matrix shows that *Local* was frequently confused with other failure types. Fig. 13(b) shows several wafer maps that were misclassified as *Local*. Although the wafer maps were misclassified, users generally accept the prediction because these wafer maps seem to saddle across the boundary of two types. This implies that the users' degree of satisfaction would likely be higher, as indicated by the overall accuracy (94.63%).

Table IV shows the comparison between the accuracy of the proposed method and that of *deep learning* and *spatial signature analysis*. *Deep learning* [28] has been demonstrated to produce state-of-the-art performance in object recognition in recent years [29]. Unlike *wafer-based clustering*, which is based on unsupervised-learning neural networks, *deep learning* is based on supervised-learning neural networks and, therefore, the results can be directly compared. In this study, we applied the well-known *deep learning* implementation method developed by Hinton [30] with default parameter settings. The results indicated that the proposed method is superior to *deep learning*. The input of deep learning was only raw wafer maps, indicating the effectiveness of the proposed features, which bear semantic meanings. In addition, the proposed method outperforms *spatial signature analysis*, indicating that the proposed semantic-bearing features are superior to the image moment features used in *spatial signature analysis*.

D. Similarity Ranking Results

Fig. 14 shows the results of two examples of WMSR. Different data sets were applied using the same queried wafer map. In Fig. 14(a), the upper left wafer map is the queried wafer map, whereas the other maps were retrieved from the data set. First, a search was performed for all *Center* wafer maps in the training set (3462 wafers) because the queried wafer map is *Center*. The figure shows that the retrieved wafer maps are similar to the queried wafer map. Next, Fig. 14(b) shows the same query with the full data set (811 457 wafers). The figure shows that the search results from the full data

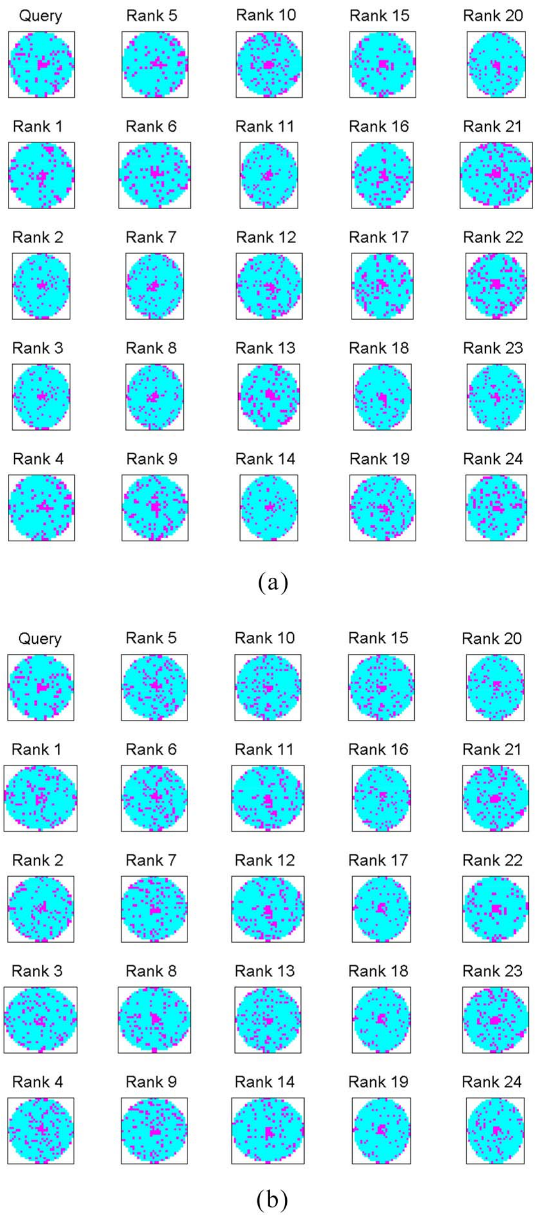


Fig. 14. WMSR results from two examples of WMSR in different scales that involved using different data sets. The upper-left image shows the queried wafer map, and the remaining wafer maps were retrieved from the corresponding data sets. Search was performed for (a) all center wafer maps in the training set (3462 wafers) and (b) the full data set (811 457 wafers).

set were comparable to those from the *Center* subset. This implies that the proposed system achieved high stability; moreover, the system successfully retrieved similar wafer maps that were unlabeled (approximately 80% of the wafer maps in the WM-811K dataset are unlabeled).

E. Computation Time

Efficiency is critical to the success of the proposed WMFPR and WMSR when deployed in fabrication. Table V lists the computation times used in each phase of the proposed method for the WMFPR. Although the feature extraction was the most time-consuming phase, the mean computation time for each wafer was 0.0737 s. The training and test phases were quite efficient as a result of the SVM. The SVM kernel evaluation in (21) accounted for most of the computation time. When parallel computing (using OpenMP library) was employed to boost the SVM kernel evaluation, the processing speed increased by a factor of approximately 3 to 4. Because the online WMFPR involves only feature extraction (0.0737 s per wafer) and SVM evaluation for prediction (0.0005 s per wafer), the proposed system can analyze more than one million wafer maps per day on a single PC ($86400/0.0742 > 10^6$). In other words, the proposed WMFPR meets the daily production requirements of a modern fabrication facility.

TABLE V
COMPUTATION TIME OF THE PROPOSED METHOD FOR WMFPR

	Set	Total time (s)	Average time (s/wafer)
Feature extraction time	Training and test sets	12749	0.0737
Training time	Training set	372	0.0068
Training time (parallel computing)	Training set	134	0.0025
Test time	Test set	272	0.0023
Test time (parallel computing)	Test set	64	0.0005

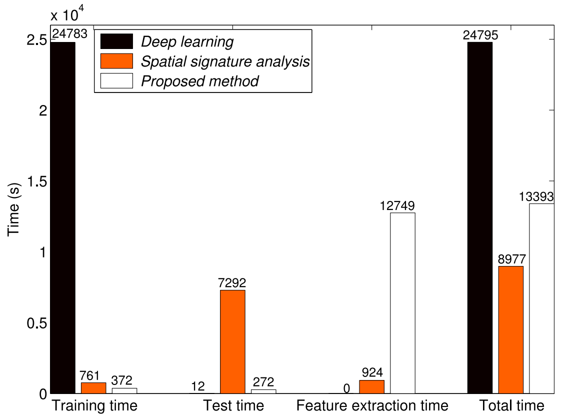


Fig. 15. Computation time comparison for WMFPR.

Fig. 15 shows the computation time comparison for WMFPR. *Deep learning* was slow when training, whereas *spatial signature analysis* was slow when testing. Furthermore, the proposed method achieved the minimal training and test time (372 + 272 = 644 s), indicating our approach has the advantage of possible parameter fine-tuning for optimizing system performance with a large-scale dataset. Because *spatial signature analysis* involved the use of simpler features (image moments) than the proposed features, it was faster than the proposed method in feature extraction time and total time, but exhibited lower accuracy. In addition, both methods were more efficient than *deep learning* regarding the total time, indicating the importance of using features for large-scale wafer map data sets.

Fig. 16 shows the WMSR search times, where the x- and y-axes show the number of wafers and search time for the top-100 similar wafer maps. The search time is estimated based on the assumption that the feature vector of each wafer map is

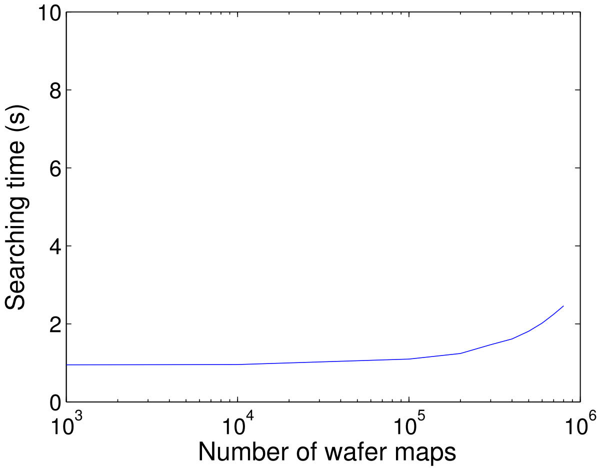


Fig. 16. WMSR search time versus the number of wafers on WM-811K data set. The top-100 similar wafer maps were obtained from 811 457 wafers in 2.5 s.

obtained in advance.[2] Fig. 16 shows that the search time was approximately 1.0 s for 100 000 wafers, and 2.5 s for 811 457 wafer maps. Based on using effective features, the proposed WMSR is highly efficient for large-scale data sets.

VII. Conclusion

In contrast to conventional approaches that generally involve applying only raw wafer maps for WMFPR and WMSR tasks, this study provides an alternative and superior method for using effective features for in-depth analysis. The proposed wafer-specific features are generally applicable to wafers of various die sizes and recipes based of the rotation- and scale-invariant design. This reduced representation is also crucial to the success of the proposed WMFPR and WMSR. The WMFPR achieved 94.63% accuracy for the test set (118 595 wafer maps). In addition, the efficiency and effectiveness of the WMSR was validated using a large-scale data set of wafer maps. The experimental results show that the time to retrieve the top-100 similar wafers from the WM-811K dataset (811 457 wafer maps) is only 2.5 s. The proposed features, combined with the classifier and similarity ranking mechanism, are effective tools for analyzing large-scale data sets.

For future research, further error analysis would assist in identifying additional robust features that are applicable to both WMFPR and WMSR. In addition, applying dimensionality reduction schemes, such as *principal component analysis* and *linear discriminant analysis*, would assist in determining whether a low-dimension feature set can obtain comparable performance. Finally, alternative machine learning techniques, such as *learning to rank* and *relevance feedback* to WMSR, should be examined.

[2] Once the fabrication has acquired a wafer map, the corresponding feature vector is extracted and stored automatically in the workflow of the TSMC. In other words, all the features of the wafer maps should be available before the similarity search begins.

References

[1] G. E. Moore, "Cramming more components onto integrated circuits," *Electronics*, vol. 38, no. 8, pp. 114–117, Apr. 1965.

[2] C. A. Mack, "Fifty years of Moores law," *IEEE Trans. Semicond. Manuf.*, vol. 24, no. 2, pp. 202–207, May 2011.

[3] TSMC. (2013). *Quarterly Results*. [Online]. Available: www.tsmc.com/uploadfile/ir/quarterly/2013/4mMqe/E/4Q13ManagementReport.pdf

[4] Q. P. He and J. Wang, "Large-scale semiconductor process fault detection using a fast pattern recognition-based method," *IEEE Trans. Semicond. Manuf.*, vol. 23, no. 2, pp. 194–200, May 2010.

[5] R. Baly and H. Hajj, "Wafer classification using support vector machines," *IEEE Trans. Semicond. Manuf.*, vol. 25, no. 3, pp. 373–383, Aug. 2012.

[6] C. F. Chen, W. C. Wang, and J. C. Cheng, "Data mining for yield enhancement in semiconductor manufacturing and an empirical study," *Expert Syst. Appl.*, vol. 33, no. 1, pp. 192–198, Jul. 2007.

[7] F. L. Chen and S. F. Liu, "A neural-network approach to recognize defect spatial pattern in semiconductor fabrication," *IEEE Trans. Semicond. Manuf.*, vol. 13, no. 3, pp. 366–373, Aug. 2000.

[8] C. F. Chen, S. C. Hsu, and Y. J. Chen, "A system for online detection and classification of wafer bin map defect patterns for manufacturing intelligence," *Int. J. Prod. Res.*, vol. 51, no. 8, pp. 2324–2338, Feb. 2013.

[9] J. Y. Hwang and W. Kuo, "Model-based clustering for integrated circuits yield enhancement," *Eur. J. Oper. Res.*, vol. 178, no. 1, pp. 143–153, Apr. 2007.

[10] T. Yuan and W. Kuo, "A model-based clustering approach to the recognition of spatial defect patterns produced during semiconductor fabrication," *IIE Trans.*, vol. 40, no. 2, pp. 93–101, 2008.

[11] T. Yuan and W. Kuo, "Spatial defect pattern recognition in semiconductor manufacturing using model-based clustering and Bayesian inference," *Eur. J. Oper. Res.*, vol. 190, no. 1, pp. 228–240, 2008.

[12] T. Yuan, S. J. Bae, and J. I. Park, "Bayesian spatial defect pattern recognition in semiconductor fabrication using support vector clustering," *Int. J. Adv. Manuf. Technol.*, vol. 51, nos. 5–8, pp. 671–683, 2010.

[13] T. Yuan, W. Kuo, and S. J. Bae, "Detection of spatial defect patterns generated in semiconductor fabrication process," *IEEE Trans. Semicond. Manuf.*, vol. 24, no. 3, pp. 392–403, Aug. 2011.

[14] K. W. Tobin, S. S. Gleason, T. P. Karnowski, S. L. Cohen, and F. Lakhani, "Automatic classification of spatial signatures on semiconductor wafer maps," in *Proc. Metrol. Insp. Process Control Microlith.*, Santa Clara, CA, USA, 1997, pp. 434–444.

[15] K. W. Tobin, S. S. Gleason, and T. P. Karnowskii, "Feature analysis and classification of manufacturing signatures based on semiconductor wafermaps," in *Proc. Mach. Vis. Appl. Ind. Insp.*, San Jose, CA, USA, 1997, pp. 14–25.

[16] T. P. Karnowski, K. W. Tobin, S. S. Gleason, and F. Lakhani, "The application of spatial signature analysis to electrical test data: Validation study," in *Proc. Insp. Process Control Microlith. XIII*, Santa Clara, CA, USA, 1999, pp. 530–540.

[17] C. H. Wang, "Recognition of semiconductor defect patterns using spatial filtering and spectral clustering," *Expert Syst. Appl.*, vol. 34, no. 3, pp. 1914–1924, 2008.

[18] T. J. Hsieh, Y. S. Huang, C. Liao, and C. F. Chien, "A new morphology-based approach for similarity searching on wafer bin maps in semiconductor manufacturing," in *Proc. 16th Int. Conf. Comput. Support. Coop. Work Design*, Wuhan, China, 2012, pp. 869–874.

[19] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Harlow, U.K.:Prentice-Hall, 2008.

[20] R. G. Keys, "Cubin convolution interpolation for digital image processing," *IEEE Trans. Audio Speech Signal Process.*, vol. 29, no. 6, pp. 1153–1159, Dec. 1981.

[21] R. M. Haralick and L. G. Shapiros, *Computer and Robot Vision*. Reading, MA, USA: Addison-Wesley, 1993.

[22] S. Cunningham and S. MacKinnon, "Statistical methods for visual defect metrology," *IEEE Trans. Semicond. Manuf.*, vol. 11, no. 1, pp. 48–53, Feb. 1998.

[23] K. P. White, B. Kundu, and C. M. Mastrangelo, "Classification of defect cluster on semiconductor wafers via Hough transform," *IEEE Trans. Semicond. Manuf.*, vol. 2, no. 2, pp. 272–278, May 2008.

[24] T. Y. Liu *et al.*, "Support vector machine with a very large-scale taxonomy," *ACM SIGKDD Explor. Newslett.*, vol. 7, no. 1, pp. 36–43, Jun. 2005.

[25] C. Cortes and V. Vapnik, "Support vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[26] C. W. Hsu, C. C. Chang, and C. J. Lin. (2014, Jul. 12). *A practical guide to support vector classification*. [Online]. Available: http://www.csie.ntu.edu.tw/∼cjlin/papers/guide/guide.pdf

[27] C. C. Chang and C. J. Lin. (2010). *LIBSVM: A library for Support Vector Machine*. [Online]. Available: http://www.csie.ntu.edu.tw/∼ cjlin/libsvm

[28] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[29] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[30] G. E. Hinton. (2014, Jul. 12). *Training a Deep Autoencoder or a Classifier on MNIST Digits*. [Online]. Available: http://www.cs.toronto.edu/∼hinton/MatlabForSciencePaper.html

Jyh-Shing R. Jang (M'93) received the Ph.D. degree in electrical engineering and computer science from the University of California, Berkeley, Berkeley, CA, USA. He studied fuzzy logic and artificial neural networks with Prof. L. Zadeh, the father of fuzzy logic. He joined MathWorks, Natick, MA, USA, and has co-authored the Fuzzy Logic Toolbox (for MATLAB). He was a Professor with the Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan, from 1995 to 2012. Since 2012, he has been a Professor with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan. His current research interests include machine learning and pattern recognition, with applications to speech recognition/assessment/synthesis, music analysis/retrieval, image identification/retrieval, and implementing industrial software for pattern recognition and computational intelligence. He has over 9000 Google Scholar citations for his seminal paper on adaptive neuro-fuzzy inference systems, published in 1993. He has published a book entitled *Neuro-Fuzzy and Soft Computing*, two books on MATLAB programming, and a book on JavaScript programming. He has also maintained toolboxes for machine learning and speech/audio signal processing and online tutorials on data clustering and pattern recognition and audio signal processing and recognition. For more information, see http://mirlab.org/jang

Ming-Ju Wu received the M.S. degree in computer science from the National Chiao Tung University, Hsinchu, Taiwan, in 2009. He is currently pursuing the Ph.D. degree in computer science from the National Tsing Hua University, Hsinchu. His research interests include information retrieval, image processing, and machine learning.

Jui-Long Chen received the M.S. degree in materials science and engineering from the National Chung Hsing University, Taichung, Taiwan, in 2004. He has been with the Taiwan Semiconductor Manufacturing Company, Hsinchu, Taiwan, since 2004, where he is currently a Technical Manager with the Manufacturing Technology Center. His research interests include developing engineering solutions for advanced technology yield enhancement and system integration functions for semiconductor manufacturing.