A Method for Wafer Defect Detection Using Spatial Feature Points Guided Affine Iterative Closest Point Algorithm

JING YANG, (Member, IEEE), YI XU, (Student Member, IEEE), HAI-JUN RONG, (Member, IEEE), SHAOYI DU, (Member, IEEE), AND HONGMEI ZHANG

1Institute of Control Engineering, Xi’an Jiaotong University, Xi’an 710049, China
2State Key Laboratory for Strength and Vibration of Mechanical Structures, School of Aerospace, Xi’an Jiaotong University, Xi’an 710049, China
3Institute of Artificial Intelligence and Robotics, Xi’an Jiaotong University, Xi’an 710049, China
4Xijing Hospital, Xi’an 710032, China

Corresponding author: Hongmei Zhang (zhm@fmmu.edu.cn)

This work was supported in part by the National Natural Science Foundation under Grant 61971343 and Grant 61773147, and in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2020JM-012.

ABSTRACT In integrated circuit manufacturing industry, in order to meet the high demand of electronic products, wafers are designed to be smaller and smaller, which makes automatic wafer defect detection a great challenge. The existing wafer defect detection methods are mainly based on the precise segmentation of one single wafer, which relies on high-cost and complicated hardware instruments. The segmentation performance obtained is unstable because there are too many limitations brought by hardware implementations such as the camera location, the light source location, and the product location. To address this problem, in this paper, we propose a method for wafer defect detection. This novel method includes two phases, namely wafer segmentation and defect detection. In wafer segmentation phase, the target wafer image is segmented based on the affine iterative closest algorithm with spatial feature points guided (AICP-FP). In wafer defect detection phase, with the inherent characteristics of wafers, a simple and effective algorithm based on machine vision is proposed. The simulations demonstrate that, with these two phases, the higher accuracy and higher speed of wafer defect detection can be achieved at the same time. For real industrial system, this novel method can satisfy the real-time detection requirements of automatic production line.

INDEX TERMS Wafer segmentation, defect detection, affine iterative closest point, corner point, machine vision.

I. INTRODUCTION

With the rapid development of semiconductor, the demand of the basic element, wafers, is growing higher and higher. Also, the quality standards of wafer is becoming stricter. There are lots of complex procedures during the manufacturing process, so it is highly possible that wafers get contaminated in the assembly line. Therefore, it is necessary to recognize the defect pattern for finding out the abnormal sources in the manufacturing process [1]. According to the research conducted by A. Freeman [2], the accuracy of human-expert based detection method is less than 45%. What’s worse, the final product contains a large number of single wafer, as shown in Figure 1, making detection much more difficult.

The associate editor coordinating the review of this manuscript and approving it for publication was Sudipta Roy.

Hence, designing a method to cut out the single wafer and recognize the defect pattern is necessary.

According to the wafer image dataset, there are two main kinds of defect pattern on the surface: spot and scratch.
The location of the spot is unknown and the angle of the scratch is uncertain and the example wafers are shown in Figure 2.

As the research hotspot, many studies on automatic defect detection have been carried out for decades and many useful methods have been proposed since then. For example, Chang et al. [3] proposed an unsupervised self-organizing neural network for automatic detection. Ng [4] revised and improved the Otsu method on common defect detection applications. Some statistic-based methods are also applied to wafer defect detection. Hwang and Kuob [5], Yuan and Kuo [6] and Wang et al. [7] proposed different probabilistic models to describe wafer defect patterns respectively. Yuan et al. [8] proposed a particle filter re-detection method via correlation filters and Tsai and Luo [9] proposed a mean shift-based method for solar wafer surface defect detection.

In recent years, the focus of wafer defect detection has been shifted to defects classification. Wang [14] proposed a spectral clustering method to recognize the defect patterns. In [15], a set of novel-rotation and scale-invariant features is presented for obtaining a reduced representation of wafer maps for future failure pattern recognition (WMFPR). Fan et al. [16] proposed a detection method based on OPTICS and multi-label classification. Bourgeat and Meriaudeau [17] used the digital holography technology and Gabor filter to extract wafer features for defect recognition. Liu et al. [18] did the research on spectral subtraction method for defect classification.

As for the wafer segmentation methods, there are many relevant researches in recent years and the existing methods can be divided into two main categories: methods based on learning and methods based on feature matching/registration.

As for matching/registration methods, there are also many existing researches. In particular, Ou et al. [10] proposed a patch-based visual method for with object selection/segmentation. Bourgeat et al. [19], [20] proposed a segmentation algorithm which is suitable for semiconductor wafer images generated by optical inspection tools. Zhen et al. [21] directly used Hough line detection method to detect wafer border and then locate the wafer based on machine learning. Pan et al. [22] proposed a binary segmentation method for locating the center line of wafer images based on the density histograms. Although these method are able to locate the wafer cutting line, the segmentation accuracy is greatly influenced by the image quality, which still cannot satisfy the requirements of detect defect. Besides, Pugazhenti and Singhai [23] proposed an image segmentation algorithm based on k-means clustering algorithm, which can automatically cluster the centroids. However, the influence of the wafer image gray scale caused by wafer defects makes this algorithm unstable for complex wafer centroid detection. Li et al. [24] proposed a LSD [25] line detection algorithm for automatic checkerboard corner extraction. Although LSD line detection algorithm is generally more accurate than Hough line detection algorithm, the results are
not as good as expected because the detection performance is greatly influenced by the image quality and border defects. It can be concluded from above discussion that the segmentation performance can be easily influenced by the wafer image quality. With unstable segmentation process, it is impossible to get expected results on the following defect detection step.

Aiming at the problems mentioned above, in this paper, we design a method combining high-precision wafer segmentation algorithm and strong real-time defect detection algorithm. In the wafer segmentation phase, we first extract spatial feature points of wafer images as guidance and then use the affine iterative closest point algorithm (AICP) for precise matching. This method, which we refer as AICP-FP, can extract important spatial feature points in wafer images and use these feature points for matching. Experiments illustrate that our proposed AICP-FP algorithm outperforms five other segmentation method. In the defect detection phase, a simple and effective detection method is designed for defect pattern recognition. This method, utilizing the inherent characteristics of the wafer image, can accomplish this detection task in a short notice. Compared with previous studies, this method greatly improves the segmentation accuracy and the real-time performance of defect detection, which can satisfy the requirements of realistic production. Furthermore, our method can reduce the cost of the whole detection system, and has a good prospect of industrial application.

The rest of the paper is organized as followers: Section 2 provides a detailed discussion of the wafer segmentation phase and defect detection phase employed in this method. Experiments and the related analysis are shown in Section 3. Finally, conclusions are drawn in Section 4.

II. PROPOSED METHOD

According to the discussion above, the schematic of this method is shown in Figure 3. This method includes two main phases: wafer segmentation and wafer defect detection. After we obtain the product wafer, the first step is to cut out the single wafer as is shown in Figure 3, then the second step is to recognize the defect pattern of the defective wafer.

A. WAFER SEGMENTATION

In wafer segmentation phase, first of all, wafer images are pre-processed using Canny edge operator and Hough transform to extract point cloud sets for registration. Also, Harris corner detection is used to extract the spatial feature points as guidance. Then, we apply the affine iterative closest point algorithm with spatial feature points guided to achieve the optimal registration between the wafer template image and wafer target image. Finally, we manage to obtain the segmented single wafer sample. The flowchart of this phase is shown in Figure 4.

1) POINT CLOUD SET EXTRACTION

We choose Canny edge detector [26] to extract edge points. As a common edge extraction operator, Canny edge operator has several advantages such as it can decrease the image noise by optimal smoothing and connect disconnected edge effectively.

The steps of Canny edge detector are summarized as follows:

1) Apply Gaussian filter to smooth the image in order to remove the noise;
2) Find the intensity gradients of the image;
3) Apply non-maximum suppression to get rid of spurious response to edge detection;
4) Apply double threshold to track the wafer image edges by hysteresis [27].

Some results of the Canny edge detector are shown in Figure 5. However, the precise registration of two point sets is based on the rough alignment of these two point sets. Therefore, appropriate initial value is an important guarantee for establishing the best match of these two point sets. Considering the fixed structures of wafer products and the regularity of product images, we use Hough Line Detection [28], [29] to detect wafer borders and we consider the intersection points \( J = \{ m_i | i = 1, 2, \ldots, N_j \} \) between borders as the initial value of next move.

A straight line equation \( y = ax + b \) in Cartesian coordinates can be expressed in Polar coordinates as follows:

\[
\rho = x \cos \theta + \sin \theta
\]
corresponding ρx
where ρ is the vertical distance from the origin to the straight
line and θ is the normal direction of this line.

The basic principle of Hough transform is to transform the
straight line in Cartesian coordinates into Polar coordinates
using the duality of points and lines. For each feature point
and discrete interval ρ, = |y| − f(x, y − f(x, y + 1)| + |f(x + 1, y) − f(x, y + 1)|;

1) Initialize Hough array (θ, ρ) with the value 0;
2) Generate a first order differential edge gradient
equation:

G(x, y) = |f(x, y) − f(x + 1, y + 1)| + |f(x + 1, y) − f(x, y + 1)|;

3) For each non-zero pixel (x, y), when G(x, y) > T, perform
the coordinate transformation ρ = xcosθ + sintθ
(where T is a threshold and we let T = 0.7Gmax(x, y)
to ensure the accuracy of edge extraction as well as
to reduce the multiplication in line detection in the experiment);
4) Add one to the counter (θ, ρ);
5) Find the maximum point, note its position, and set the
Hough array back to 0;
6) Repeat step 5 until find peaks required;
7) Calculate the slope of the line k = (y2 − y1)/(x2 − x1)
according to the corresponding point of each line;
8) Connect the straight lines which share the same slope
and the same point and establish a point set of
intersections.

Hough line detection has stronger anti-noise ability than
other methods and it can reduce the impact of wafer border
flaws due to its discontinuous insensitivity to the edge. Hough
line detection based on edge gradient can effectively reduce
the multiplication and accelerate the detection speed.

Finally, we take the border points as point cloud sets for
next registration.

2) AFFINE ITERATIVE CLOSEST POINT ALGORITHM WITH
SPATIAL FEATURE POINTS GUIDED

Given the point cloud set of wafer target image P = {p|i}Np
and wafer template image Q = {q|i}Nq, we manage to solve
the matching problem based on the shape registration. The
similarity between the wafer target image and the wafer
template image is measured by the error distance between
those two point cloud sets. In another word, we need to find a
transformation matrix T which is an optimal match between
the wafer target feature point set and the wafer template
feature point set. Therefore, we can write the objective based
on the least square (LS) criterion as follows:

\[
\min_{T, \theta, \rho} \sum_{i=1}^{Np} \| T(p_i) - q_j \|^2, \quad i = 1, 2, \ldots, N_p
\]  

(2)

In (2), the correspondence between the wafer target point set
and the wafer template point set is \( \{(\tilde{p}_i, \tilde{q}_j)\}_{i=1}^{N_p} \) and assume
the optimal matching relation is \( \{(\tilde{p}_i, \tilde{q}_j)\}_{i=1}^{N_p} \).

The matching process between those two point sets is
actually the affine transformation process, which is so called
shape registration. First of all, the transformation matrix T
needs to be linear transformed, so (2) is written as:

\[
\min_{A, \tilde{T}} \sum_{i=1}^{N_p} \left\| (A\tilde{p}_i + \tilde{T}) - \tilde{q}_j \right\|^2
\]

\[s.t. \quad \det(A) \neq 0\]  

(3)

In (3), T is decomposed into a full rank matrix and a
translation vector, where is a rotation matrix from point set
to point set.
The traditional affine iterative closet algorithm \([31]\) solves the problem of optimization (3) by iteration. In each iteration, the algorithm consists following two steps:

1) Set up the correspondence \(\{(i, c_k(i))\}_{i=1}^{N_p}\) between point set \(P\) and \(Q\) according to the affine transformation \((A_{k-1}, t_{k-1})\) at the \((k-1)\)th step:

\[
c_k(i) = \arg \min_{j \in \{1, \ldots, N_q\}} \left( \| (A_{k-1}\tilde{p}_i + \tilde{t}_{k-1}) - \tilde{q}_j \|^2 \right) \quad i = 1, 2, \ldots, N_p
\]

(4)

2) Calculate the \(k\)th transformation \((A_k, t_k)\) based on the current correspondence \(\{(i, c_k(i))\}_{i=1}^{N_p}\):

\[
(A_k, t_k) = \arg \min_{\text{det}(A) \neq 0, \tilde{t}} \left( \sum_{i=1}^{N_p} \| (A\tilde{p}_i + \tilde{t}) - \tilde{q}_{c_k(i)} \|^2 \right)
\]

(5)

Repeat the above two steps until the mean square error between the two point sets reaches the minimum or the maximum number of iterations is reached, we obtain the final affine transformation \((A_k, t_k)\).

The traditional affine iterative closest point algorithm can achieve fast and efficient registration between two point sets, but in the absence of constraints, the affine registration is easy to fall into local optimum. Based on the strong local feature of wafer images, in addition to avoid falling into local optimum, we reset up the objective function guided by spatial feature points \([32]\). Harris corner detection \([33]\) is used to extract spatial feature point sets \(X\) and \(Y\) from the target point set \(P = \{\tilde{p}_i\}_{i=1}^{N_p}\) and template point set \(Q = \{\tilde{q}_j\}_{i=1}^{N_q}\) respectively.

Then we use the traditional affine iterative closest point algorithm to pre-register these two spatial feature point sets and obtain the corresponding sets \(X = \{\tilde{x}_i\}_{i=1}^{N_l}\) and \(Y = \{\tilde{y}_l\}_{l=1}^{N_l}\).

We introduce the spatial feature information to (3) to reset up the new objective function, which is shown as follows:

\[
\min_{A, \tilde{t}, \tilde{c}_k(i) \in 1, \ldots, N_q} \left( \frac{1}{N_p} \sum_{i=1}^{N_p} \| (A\tilde{p}_i + \tilde{t}) - \tilde{q}_{c_k(i)} \|^2 \right) + \frac{\alpha}{N_l} \sum_{l=1}^{N_l} \| (A\tilde{x}_l + \tilde{t}) - \tilde{y}_l \|^2
\]

(6)

where \(\alpha\) is the weight coefficient of the spatial feature points, \(\tilde{x}_l\) and \(\tilde{y}_l\) is the corresponding spatial feature points in the two feature point sets, \(N_p\) and \(N_l\) are the number of points in the target point set and the feature point set respectively.

The first step of the algorithm can be accomplished by some fast algorithms such as the nearest point search based on Delaunay tessellation \([34]\) or \(k - d\) tree \([35]\).

Therefore, the key point is to solve the second step in the iteration. The (7) then becomes:

\[
F(A, \tilde{t}) = \frac{1}{N_p} \sum_{i=1}^{N_p} \| A\tilde{p}_i + \tilde{t} - \tilde{q}_{c_k(i)} \|^2 + \frac{\alpha}{N_l} \sum_{l=1}^{N_l} \| A\tilde{x}_l + \tilde{t} - \tilde{y}_l \|^2
\]

(8)

We can derive the objective function \(F(A, \tilde{t})\) with respect to \(\tilde{t}\):

\[
\frac{dF(A, \tilde{t})}{d\tilde{t}} = \frac{2}{N_p} \sum_{i=1}^{N_p} (A\tilde{p}_i + \tilde{t} - \tilde{q}_{c_k(i)})^2 + \frac{2\alpha}{N_l} \sum_{l=1}^{N_l} (A\tilde{x}_l + \tilde{t} - \tilde{y}_l)^2
\]

(9)

Let \(dF(A, \tilde{t})/d\tilde{t} = 0\), we can get:

\[
\tilde{t} = \frac{1}{1 + \alpha} \left( \frac{\sum_{i=1}^{N_p} \tilde{q}_{c_k(i)} + \alpha \sum_{l=1}^{N_l} \tilde{y}_l - A \sum_{i=1}^{N_p} \tilde{p}_i - a A \sum_{l=1}^{N_l} \tilde{x}_l}{N_p} \right)
\]

(10)

Take \(\tilde{t}\) into (8), we can simplify the \(F(A, \tilde{t})\) and get:

\[
\tilde{p}_i = \left( \frac{1}{N_p} \sum_{i=1}^{N_p} \tilde{p}_i + \frac{\alpha}{N_l} \sum_{l=1}^{N_l} \tilde{x}_l \right) \frac{1}{1 + \alpha}
\]

\[
\tilde{q}_{c_k(i)} = \left( \frac{1}{N_p} \sum_{i=1}^{N_p} \tilde{q}_{c_k(i)} + \frac{\alpha}{N_l} \sum_{l=1}^{N_l} \tilde{y}_l \right) \frac{1}{1 + \alpha}
\]

\[
\tilde{x}_l = \left( \frac{\alpha}{N_l} \sum_{l=1}^{N_l} \tilde{x}_l \right) \frac{1}{1 + \alpha}
\]

\[
\tilde{y}_l = \left( \frac{\alpha}{N_l} \sum_{l=1}^{N_l} \tilde{y}_l \right) \frac{1}{1 + \alpha}
\]

(11)

Then, the objective function \(F(A, \tilde{t})\) can be written as:

\[
F(A) = \sum_{i=1}^{N_p} \| A\tilde{p}_i - \tilde{q}_{c_k(i)} \|^2 + \sum_{l=1}^{N_l} \| A\tilde{x}_l - \tilde{y}_l \|^2
\]

(12)
Let \( N = N_p + N_l \), \( E = e_j^N \), \( F = f_j^N \), define \( e_j \) and \( f_j \) as follows:

\[
\vec{e}_j = \begin{cases} 
\vec{p}_j^1 & 1 \leq j \leq N_p \\
\vec{x}_{j-N_p} & N_p + 1 \leq j \leq N 
\end{cases}
\]

\[
\vec{f}_j = \begin{cases} 
\vec{q}_{(i)}^j & 1 \leq j \leq N_p \\
\vec{y}_{j-N_p} & N_p + 1 \leq j \leq N 
\end{cases}
\]

(13)

The (12) can be simplified as:

\[
F(A) = \sum_{i=1}^{N} \left\| \vec{A}\vec{e}_j - \vec{f}_j \right\|^2_2
\]

(14)

In order to minimize the objective function \( F(A) \), we make \( dF(A)/dA = 0 \):

\[
\sum_{j=1}^{N} (A\vec{e}_j - \vec{f}_j)\vec{e}_j^T = 0
\]

(15)

and the affine matrix \( A \) can be obtained as follows:

\[
A = \left( \sum_{j=1}^{N} \vec{f}_j\vec{e}_j^T \right) \left( \sum_{j=1}^{N} \vec{e}_j\vec{e}_j^T \right)^{-1}
\]

(16)

Take affine matrix \( A \) into (10), we can obtain the translation vector \( \vec{t} \). To evaluate the results of the registration between two point sets, an error function is set up as follows:

\[
e_k = \frac{1}{N_p} \sum_{i=1}^{N_p} \left\| A_k\vec{p}_i + \vec{t}_k - \vec{q}_{(i)} \right\|^2_2
\]

\[
+ \frac{\alpha}{N_l} \sum_{l=1}^{N_l} \left\| A_k\vec{x}_l + \vec{t}_k - \vec{y}_l \right\|^2_2
\]

(17)

In the \( k \)th iteration, if the error function reaches the minimum value \( |e_k - e_{k-1}| \leq \delta \) or the maximum number of iterations is reached, the iteration stops and outputs the final affine transformation \( (A_k, \vec{t}_k) \).

The whole process of affine iterative closest point algorithm with spatial feature points guided is summarized as follows:

3) AICP ALGORITHM WITH SPATIAL FEATURE POINTS GUIDED

1) Obtain point cloud set of wafer target image \( P = \{\vec{p}_i\}_{i=1}^{N_p} \), wafer template image \( Q = \{\vec{q}_i\}_{i=1}^{N_l} \). Two spatial feature point sets \( X = \{\vec{x}_i\}_{i=1}^{N_l} \) and \( Y = \{\vec{y}_i\}_{i=1}^{N_l} \);
2) Initialize affine matrix \( A_0 \), translation vector \( \vec{t}_0 \), spatial feature points weight \( \alpha \);
3) Find the correspondence \( \{\vec{p}_i, \vec{q}_{(i)}\} \) between two point sets using (4) based on the affine transformation of the \( (k-1) \)th step;
4) Calculate the \( k \)th transformation \( (A_k, \vec{t}_k) \) using (10) and (16);
5) Repeat step 3 and step 4 until \( |e_k - e_{k-1}| \leq \delta \) or the maximum number of iterations is reached.
6) Output the affine matrix \( A_k \) and the translation vector \( \vec{t}_k \).

Remark 1: The computational complexity of the AICP-FP algorithm is the same as that of the traditional AICP algorithm. The whole process of the AICP-FP algorithm is similar to the traditional AICP. Both of them contains two main steps, the first step is to establish the correspondence, the second step is to calculate the transformation. As the transformation is the closed-form solution, the computing time is negligible compared with the first step. So, the whole computing time focus on the time of establishing the correspondence, where the computational complexity is \( O(N_p \ln N_q) \) [36].

Theorem 1: The AICP-FP algorithm converges monotonically to a local minimum with respect to the mean square error from any given initial transformation.

Proof: Given two point sets \( P = \{\vec{p}_i\}_{i=1}^{N_p} \) and \( Q = \{\vec{q}_i\}_{i=1}^{N_l} \). The initial affine matrix is \( A_0 \) and the initial translation vector is \( \vec{t}_0 \). At the \( (k-1) \)th step, assume the \( (A_{k-1}, \vec{t}_{k-1}) \) is known, the corresponding closest point set \( \{\vec{p}_i, \vec{q}_{c(i)}\} \) is computed. Then the mean square error (MSE) is shown as follows:

\[
e_k = 1 \frac{1}{N_p} \sum_{i=1}^{N_p} \left\| A_{k-1}\vec{p}_i + \vec{t}_{k-1} - \vec{q}_{c(i)} \right\|^2_2
\]

\[
+ \frac{\alpha}{N_l} \sum_{l=1}^{N_l} \left\| A_{k-1}\vec{x}_l + \vec{t}_{k-1} - \vec{y}_l \right\|^2_2
\]

(18)

By minimizing (7), we can obtain the \( k \)th transformation \( (A_k, \vec{t}_k) \). Therefore, the MSE is shown as follows:

\[
e_k = 1 \frac{1}{N_p} \sum_{i=1}^{N_p} \left\| A_k\vec{p}_i + \vec{t}_k - \vec{q}_{c(i)} \right\|^2_2
\]

\[
+ \frac{\alpha}{N_l} \sum_{l=1}^{N_l} \left\| A_k\vec{x}_l + \vec{t}_k - \vec{y}_l \right\|^2_2
\]

(19)

It is apparent that \( e_k \leq e_k \), otherwise, if \( A_k = A_{k-1}, \vec{t}_k = \vec{t}_{k-1} \), namely the algorithm does converges, there is \( e_k = e_k \).

Next, assume the \( (A_k, \vec{t}_k) \) is known, we can compute the corresponding closest point set \( \{\vec{p}_i, \vec{q}_{c(i)}\} \) via minimizing (4). Therefore, the MSE is shown as follows:

\[
e_{k+1} = 1 \frac{1}{N_p} \sum_{i=1}^{N_p} \left\| A_k\vec{p}_i + \vec{t}_k - \vec{q}_{c(i)} \right\|^2_2
\]

\[
+ \frac{\alpha}{N_l} \sum_{l=1}^{N_l} \left\| A_k\vec{x}_l + \vec{t}_k - \vec{y}_l \right\|^2_2
\]

(20)

Since the \( e_k \) is the minimum of (4), there is \( e_{k+1} \leq e_k \).
Accordingly, repeat the procedures above for any \( k \) and we can obtain:

\[
0 \leq \ldots \leq e_{k+1} \leq e_k \leq e_k \leq \ldots \leq e_1
\]

(21)

According to the Monotonic Sequence Theorem “Every bounded monotonic sequence of real numbers is convergent”, the AICP-FP algorithm converges monotonically to the local minimum with respect to the mean square error.
B. DEFECT DETECTION

In wafer defect detection phase, first of all, we do the subtraction between the target wafer image and the qualified flawless one. Next, we calculate the peak value of the wafer image gray histogram and distinguish the defective wafer from the flawless wafer with a threshold $T_1$. After obtaining the defective wafer, we binary it and do the close operation to extract the inherent characteristics. In the end, we calculate the major axis length of the connected domain to distinguish the scratch defect from the spot defect with another threshold $T_2$. The flowchart of this phase is shown in Figure 6.

1) SUBTRACTION BETWEEN TWO GRAY-SCALE IMAGES

After obtaining the single wafer image, the first step is to distinguish between the defective wafer and the qualified flawless wafer. Considering the wafer image features, we propose a simple and efficient strategy using the image histogram.

An image histogram is a type of histogram that acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. The equation for calculating the frequency $P(b_k)$ of gray-scale value $b_k$ is given by:

$$P(b_k) = \frac{N_k}{M}, \quad k = 0, 1, \ldots, (L - 1)$$  \hspace{1cm} (22)

where $M$ is the number of all the pixels in the image, $N_k$ is the number of pixels whose gray scale value are, $L$ is the gray level, normally $L = 256$.

It is obvious that the defective wafer image has certain scale difference in some regions, in order to highlight this difference, we do the subtraction between the target image and the qualified flawless one and the difference will emerge in gray scale histogram. Apparently, if the target image has no defect, the subtraction result will be almost all 'black' because the gray value of the corresponding pixels is similar, which means there is a very high peak frequency in low gray level. According to this, we calculate the peak frequency and set an appropriate threshold $T_1$ to distinguish defective wafers from flawless ones. If the peak value is greater than the threshold $T_1$, it means that the target wafer is flawless, otherwise, the target wafer is defective. Experiments show that the peak frequency of flawless wafer images is greater than 10000 and the peak frequency of defective wafer images is smaller than 3000. Therefore, we set the threshold $T_1 = 5000$.

2) BINARIZATION USING 2-MODE METHOD

There are two main defect patterns shown in Figure 2: spot and scratch. The most difference between the spot defect and the scratch defect is that the scratch has obvious linear feature. It is natural to distinguish the scratch from the spot using this inherent feature. So, extracting the scratch information has become the crucial step in this defect detection phase.

As the interesting region which contains the defect has a sharp contrast to the image background, the gray histogram shows two peaks which are formed by the interesting region and the background respectively. Based on this, we use the 2-mode method [37] to obtain the binary image. The 2-mode method choose the trough value of the gray histogram as the threshold for image binarization.

3) SCRATCH CONNECTION USING CLOSE OPERATION

However, it is highly possible that the scratch is discontinuous in the binary image because the scratch itself is discontinuous or its distribution of gray scale is uneven. The discontinuity of the scratch can weaken the linear feature and inevitably affect the performance of defect detection. So, we need to connect the scratch to enhance the linear feature. There are several common methods for scratch connection such as Hough transform and fitting line with least square method [38].

However, Hough transform involves space conversion and abundant computation, which takes too much time. As for the fitting line with least square method, there are too many regions in the image that can be detected as lines, which is
inappropriate in this case. In order to improve the accuracy and save the computing time, we choose to use close operation [37] to connect the scratch. As a simple image processing operation, the equation of close operation is given by:

$$A \bullet B = (A \oplus B) \ominus B$$

(23)

where A is a binary image, B is a structuring element, $\oplus$ and $\ominus$ denote the dilation and erosion, respectively.

The close operation is used to fill small holes in an object, to connect adjacent objects and to smooth its boundary, while it does not significantly change its area, which is suitable in this case.

After the close operation is finished, we calculate the major axis length of the connected domain and set a threshold $T_2$ to distinguish the scratch from the spot. Clearly, the length of the scratch is much longer than the spot’s and the two lengths are not the same order of magnitude. Experiments shows that setting the threshold $T_2 = 15$ has best performance.

All steps can be summarized as:

- step 1: Do the subtraction between the gray histogram of the wafer target image and the qualified flawless one;
- step 2: Calculate the peak frequency of the result in step 1, if the peak value is greater than the threshold $T_1$, it means the target wafer is flawless, otherwise, it is defective;
- step 3: Using 2-mode method for the defective image binaryization;
- step 4: Apply the close operation to the binary image in step 3 for scratch connection;
- step 5: Calculate the major axis length of the connected domain. If the length is greater than the threshold $T_2$, it means the target wafer has the scratch defect, otherwise, it has the spot defect.

### III. RESULTS AND ANALYSIS

The whole experiment includes two main parts: wafer segmentation and defect detection. In order to verify the accuracy and robustness of the proposed method, we first propose an evaluation metric for wafer segmentation and then make comparative experiments. After wafer segmentation accomplished, we calculate the accuracy and running time in defect detection part, some intermediate results are demonstrated using example wafer images.

The experimental datasets are acquired in industrial process and an example of this wafer product image is shown in Figure 7, the total number is 638 and it includes both flawless wafers and defective wafers. The experimental environment is Intel(R) Core(TM) i7-8700 @ 3.20GHz with 8GB RAM.

#### A. WAFER SEGMENTATION

In this method, wafer segmentation is the essential prerequisite and the precise wafer segmentation results are crucial to the following defect detection. In order to verify the accuracy of our proposed wafer segmentation method, we make comparative experiments with five other wafer segmentation methods: segmentation based on the size, segmentation based on line segment detector (LSD) algorithm, centroid segmentation based on k-means clustering algorithm, traditional affine iterative closest algorithm and corner points detection algorithm.

1) EVALUATION METRIC

Since there is no existing evaluation metric for wafer segmentation, in order to verify wafer segmentation performance, we propose a new evaluation metric based on the wafer structure. This evaluation metric is illustrated in Figure 8.

In Figure 8, let the ideal segmentation edges are surrounded by P1, P2, P3, P4 and the actual edges are surrounded by P1’, P2’, P3’, P4’. Let the point set $Z = \{p_i, i = 1, 2, 3, 4\}$ and $Z' = \{p_i', i = 1, 2, 3, 4\}$, where $\bar{p}_i = (x_i, y_i)$ and $\bar{p}_i' = (x_i', y_i')$. We compute the Euclidean distance between those corresponding points to represent the segmentation error $E$.

$$E = \sum_{i=1}^{4} \left\| \bar{p}_i' - \bar{p}_i \right\|^2$$

(24)

The smaller the $E$ is, the better the segmentation performance is, vice versa.

2) RESULTS

We number those 638 wafers from left top to right bottom and then make experiments using six wafer segmentation methods respectively. We run each algorithm for 50 times and then calculate the running time for each sample. We select...
TABLE 1. Results of six wafer segmentation methods.

| Wafer | Size (mm) | LSD Centroid | AICP | Corner Points | AICP-FP |
|-------|-----------|--------------|------|---------------|---------|
| E89   | 86.163    | 28.797       | 20.255 | 27.672        | 191.656  | 4.549   |
| E121  | 128.062   | 22.598       | 16.987 | 21.023        | 153.856  | 4.375   |
| E253  | 175.317   | 31.335       | 24.598 | 22.748        | 93.736   | 2.812   |
| E154  | 187.446   | 24.068       | 22.850 | 24.537        | 28.129   | 2.584   |
| E333  | 350.907   | 32.642       | 36.020 | 41.343        | 235.570  | 2.156   |
| E366  | 364.878   | 47.778       | 36.906 | 46.569        | 137.058  | 2.806   |
| E457  | 475.714   | 39.722       | 40.232 | 54.517        | 74.691   | 3.072   |
| E652  | 573.606   | 45.045       | 39.125 | 25.747        | 84.596   | 2.722   |
| Average | 292.762   | 34.000       | 29.622 | 33.020        | 127.160  | 3.125   |

| Time (s) | 0.001s | 0.4s | 0.2s | 0.8s | 0.4s | 0.9s |

For our proposed AICP-FP algorithm, its running time is the longest but less than 1s. Considering its outstanding performance, it is worthwhile during the segmentation process. We will focus on how to improve the operating efficiency of the algorithm in our future work.

In order to present the segmentation effect directly, the sample 89 with spots and the sample 125 with serious scratches are shown for qualitative analysis. It is obvious that the error of the segmentation method based on the size in Figure 9(a) and Figure 10(a) is the largest. The segmented wafer is not complete at all. We can see from Figure 9(b) and Figure 9(b), the right and bottom borders are not complete either. As for the centroid segmentation method and the segmentation method based on the AICP algorithm, the wafer sample edge (red line) in Figure 9(c)-9(d) and Figure 10(c)-10(d) basically coincides with the template edge (green line). However, in Figure 9(c) and Figure 10(c), the outer edges such as the borders deviate farther from the edges of the template. The middle parts...
Segmentation effect images of wafer sample 125.

Segmentation based on the size.

Segmentation based on LSD algorithm.

Segmentation based on k-means clustering algorithm.

Segmentation based on traditional AICP algorithm.

Segmentation based on corner points

Segmentation based on AICP-FP algorithm.

FIGURE 10. Segmentation effect images of wafer sample 125.

Figure 10 shows the segmentation effect images of wafer sample 125. The segmentation based on the AICP algorithm are registered better than those using centroid segmentation method. We can see from Figure 9(e) and Figure 10(e), there are some feature points in the left wafer image linked to the points in the similar but wrong part of the right wafer image, which leads to the bad error in Table 1. As for our proposed method, Figure 9(f) and Figure 10(f) have fewer red lines especially the left part in Figure 9(f) and the lower right part in Figure 10(f), which means the error is smaller. The results are consistent with the errors in Table 1. In conclusion, our proposed segmentation method based on the AICP-FP algorithm outperforms other methods.

3) VALIDATION
In order to validate our proposed segmented method, we use another kind of wafer samples for validation. The example of this wafer image is shown in Figure 11. The same as above, we number these wafer samples from left top to right bottom and then make experiments using six wafer segmentation methods respectively. The segmentation error $E_i$ are shown in Table 2.

As can be seen in Table 2, the results of the method based on size are still the worst because of the accumulative error. The error of the method based on corner points is large and still unstable because of the mismatches. Because of the asymmetry of the wafer sample shape, the average error of the centroid segmentation method is 13.136, which the third large among six methods. The performance of the method based on LSD is better than the method based on AICP because the straight lines in wafer sample are clear and easy to be detected. To sum up, our proposed AICP-FP method is still the best among six segmentation methods, which means our method is robust to different kinds of wafer samples.

B. WAFER DEFECT DETECTION
In this section, we make experiments on the proposed wafer defect detection method the segmented wafer image. Some intermediate results of this phase as well as final results are shown in this section.

1) RESULTS
Several example wafer images are employed to validate the performance of the defect detection method. Figure 12 shows the subtraction results and their gray histograms using the three wafer images in Figure 2.

As is shown in Figure 12, if the target wafer image is flawless, the subtraction result is almost all black which leads to a quite high peak in its gray histogram, so we use this characteristic to distinguish the defective wafer from the flawless wafer. As for the defective wafer image, the defect emerges in the subtraction result image, which contrasts the
wafer image background. It can be seen from the gray histogram in Figure 12(b) and 12(c) that the gray histogram of the subtraction result has the two-peak characteristic.

We also use the common region growing algorithm [39] to make comparative experiments. Figure 13 shows the results of region growing algorithm applied to the same defective wafer image with different seed points. The left image in Figure 13(a) and 13(b) is the same defective image where the seed point is starred differently and the right image is the result of using region growing algorithm.

As is shown Figure 13, the results are different with different seed points. If we choose the seed point as starred in Figure 13(a), the result shows the complete scratch but only the contour. If we choose another seed point as starred in Figure 13(b), we cannot obtain the whole scratch. In fact, including the artificial selection of seed point, the region growing algorithm is complex and not ideal in this case.

In Figure 14, the first column are some example wafer images, the results of image binaryzation using 2-mode method are shown in the second column, the results of applying close operation are shown in the last column.

We can see from the second column in Figure 14, the defects emerge in binary images and the scratches in Figure 14(b) and 14(c) are discontinuous. In the third column, after applying the close operation, the discontinuous scratches get connected effectively. In Figure 14(d), some of these spots also get connected because they are too close to be considered as different parts of one same scratch, however, even those spots get connected, its length is still shorter than the length of the actual scratch by over an order of magnitude.
TABLE 3. Results of wafer defect detection.

|                      | step 1 | step 2 | step 3 | step 4 and 5 | step 1–5 |
|----------------------|--------|--------|--------|--------------|----------|
| Time                 | 0.08s  | 0.027s | 0.1s   |              |          |
| Recognition Rate     | 100%   | 94.15% | 94.15% |              |          |

TABLE 4. Results of defective wafer detection.

|                | Scratches | Spots | Total |
|----------------|-----------|-------|-------|
| Number         | 171       | 154   | 325   |
| Recognition Number | 164       | 142   | 306   |
| Recognition Rate | 95.91%    | 92.21%| 94.15%|

Comparing the results of close operation in Figure 14 and the results of region growing algorithm in Figure 13, it is obvious that the close operation is much more efficient in this case.

In this experiment, we calculate the average running time and the detection accuracy of the steps mentioned in the above defect detection phase (II.B). The results are shown in Table 2.

The step 1 and step 2 are designed to distinguish the defective wafer from flawless one and the following three steps are designed to identify the defect patterns, so we calculate the average running time of these two major components, respectively.

We run step 1 and step 2 for 50 times and the running time is 0.08s. We also run step 3, 4 and 5 for 50 times and the running time of is 0.027s. The running time is quite short which can save a lot of time in actual production.

There are 638 wafer sample, including 313 flawless samples and 325 defective ones (171 samples with scratches and 154 samples with spots). The accuracy of the step 1 and 2 is 100%, which means all the defective wafer samples can be distinguished correctly. As for defective wafer samples, 164 out of 171 samples with scratches (95.91%) and 142 out of 154 samples with spots (92.21%) can be correctly identified. In total 306 samples and the recognition rate is 94.15%.

At last, we combine these two phases as a whole method and test on it. We run this whole method for 50 times and the average running time of the whole method is about 1s, which is fairly fast enough to be deployed in the industrial assembly lines.

IV. CONCLUSION

A method is proposed for wafer segmentation and defect detection in this paper. This method includes two main functions: cutting out the single wafer and recognizing the defect pattern. Aiming at the problems among the existing methods, we propose a wafer segmentation method based on the AICP-FP algorithm without relying on hardware facilities as well as a defect detection method based on machine vision. The details are as follows:

1) The proposed segmentation algorithm based on AICP-FP outperforms the five other segmentation algorithms. The running time is acceptable considering its best performance.
2) The proposed defect detection algorithm is much more accurate than human-expert based detection algorithm.

Its running time is 0.1s, which is fast enough for industrial assembly line.

Combining these two parts as a whole, its accuracy and the running time is outstanding which can satisfy the requirements of realistic production. The research of this paper provides important technical reference for reducing manufacturing costs and improving the automation level.

REFERENCES

[1] 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.
[2] A. Drozda-Freeman, M. McIntyre, M. Retersdorf, C. Wooten, X. Song, and A. Hesse, “The application and use of an automated spatial pattern recognition (SPR) system in the identification and solving of yield issues in semiconductor manufacturing,” in Proc. IEEE/SEMI Adv. Semiconductor Manuf. Conf., Jun. 2007, pp. 302–305.
[3] C.-Y. Chang, J.-W. Chang, and M. der Jeng, “An unsupervised self-organizing neural network for automatic semiconductor wafer defect inspection,” in Proc. IEEE Int. Conf. Robot. Autom., Apr. 2006, pp. 3000–3005.
[4] H.-F. Ng, “Automatic thresholding for defect detection,” Pattern Recognit. Lett., vol. 27, no. 14, pp. 1644–1649, Oct. 2006.
[5] J.-Y. Hwang and W. Kuo, “Model-based clustering for integrated circuit yield enhancement,” Eur. J. Oper. Res., vol. 178, no. 1, pp. 143–153, Apr. 2007.
[6] T. Yuan and W. Kuo, “A model-based clustering approach to the recognition of the spatial defect patterns produced during semiconductor fabrication,” IEEE Trans., vol. 40, no. 2, pp. 93–101, Nov. 2007.
[7] C.-H. Wang, W. Kuo, and H. Bentoumi, “Detection and classification of defect patterns on semiconductor wafers,” IEEE Trans. Electron. Imag., vol. 38, no. 12, pp. 1059–1068, Dec. 2006.
[8] D. Yuan, X. Du, D. Li, Y. Liang, and X. Zhang, “Particle filter re-detection for visual tracking via correlation filters,” Multimedia Tools Appl., vol. 78, no. 11, pp. 14277–14301, Jun. 2019.
[9] D.-M. Tsai and J.-Y. Luo, “Mean shift-based defect detection in multycrystalline solar wafer surfaces,” IEEE Trans. Ind. Informat., vol. 7, no. 1, pp. 125–135, Feb. 2011.
[10] W. Ou, D. Yuan, D. Li, B. Liu, D. Xia, and W. Zeng, “Patch-based visual tracking with online representative sample selection,” J. Electron. Imag., vol. 26, no. 3, May 2017, Art. no. 033006.
[11] J. Long, E. Shehhammer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 4, pp. 640–651, Apr. 2017.
[12] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent., 2015, pp. 234–241.
[13] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834–848, Apr. 2018.
[14] C. H. Wang, “Recognition of semiconductor defect patterns using spectral clustering,” Expert Syst. Appl., vol. 34, no. 3, pp. 1914–1923, 2008.
[15] M.-J. Wu, J.-S.-R. Jang, and J.-L. Chen, “Wafer map failure pattern recognition and similarity ranking for large-scale data sets,” IEEE Trans. Semicond. Manuf., vol. 28, no. 1, pp. 1–12, Feb. 2015.
[16] M. Fan, Q. Wang, and B. van der Waal, “Wafer defect patterns recognition based on OPTICS and multi-label classification,” in Proc. IEEE Adv. Inf. Manage., Commun., Electron. Autom. Control Conf. (IMCEC), Oct. 2016, pp. 912–915.
[17] P. Bourgeat and F. Meriaudeau, “Classifier vote and Gabor filter banks for wafer segmentation,” in Proc. IEEE Int. Conf. Image Process., Sep. 2005, pp. 1–4.
[18] H. Liu, W. Zhou, Q. Kuang, L. Cao, and B. Gao, “Defect detection of IC wafer based on spectral subtraction,” IEEE Trans. Semicond. Manuf., vol. 23, no. 1, pp. 141–147, Feb. 2010.
[19] P. Bourgeat, F. Meriaudeau, and K. W. Tobin, “Content-based segmentation of patterned wafer for automatic threshold determination,” Proc. SPIE, vol. 5011, pp. 183–189, May 2003.
[20] F. Meriaudeau, “Content based segmentation of patterned wafers,” J. Electron. Imag., vol. 13, no. 3, pp. 428–435, Jul. 2004.
[21] Z. Zhen, D. Shuguang, and M. Ping’an, “Wafer defects detecting and classifying system based on machine vision,” in Proc. 8th Int. Conf. Electron. Meas. Instrum., Aug. 2007, pp. 520–523.

[22] N. Pan, R. Liu, and M. Wang, “Locating the centre line of paddle vats for cutting wafer images by using binary segmentation,” in Proc. 5th Int. Symp. Distrib. Comput. Appl. Bus., Eng. Sci., Aug. 2010, pp. 561–564.

[23] A. Pugazhenthi and J. Singhai, “Automatic centroids selection in K-means clustering based image segmentation,” in Proc. Int. Conf. Commun. Signal Process., Apr. 2014, pp. 1279–1284.

[24] H. Li, X.-M. Zhang, and Z. Chen, “Automatic corner detection of checkerboard based on LSD,” Opt. Precis. Eng., vol. 23, no. 12, pp. 3480–3489, 2015.

[25] R. Grompone Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, “LSD: A line segment detector,” Image Process. Line, vol. 2, pp. 35–55, Mar. 2012.

[26] R. Biswas and J. Sil, “An improved candi edge detection algorithm based on Type-2 fuzzy sets,” Procedia Technol., vol. 4, no. 11, pp. 820–824, 2012.

[27] A. A. Adly and S. K. Abdelhafiz, “Implementation of magnetostriction Preisach-type models using orthogonally coupled hysteresis operators,” Phys. B, Condens. Matter, vol. 403, nos. 2–3, pp. 425–427, 2008.

[28] N. Aggarwal and W. C. Karl, “Line detection in images through regularized Hough transform,” IEEE Trans. Image Process., vol. 15, no. 3, pp. 582–591, Mar. 2006.

[29] D. Duan, M. Xie, Q. Mo, Z. Han, and Y. Wan, “An improved Hough transform for line detection,” in Proc. Int. Conf. Comput. Appl. Syst. Modeling (ICCASM), Oct. 2010, p. V2-354.

[30] Q. I. Bao-Ying, “Improve efficiency of line detection using Hough transform,” Comput. Syst. Appl., no. 6, pp. 228–231, 2012.

[31] S. Du, N. Zheng, and S. Ying, “ICP with bounded scale for registration of M-D point sets,” in Proc. IEEE Int. Conf. Multimedia Expo, Jul. 2007, pp. 1291–1294.

[32] S. Du, W. Cui, L. Wu, S. Zhang, X. Zhang, G. Xu, and M. Xu, “Precise iterative closest point algorithm with corner point constraint for isotropic scaling registration,” Multimedia Syst., vol. 25, no. 2, pp. 119–126, Apr. 2019.

[33] W. Wang, Y. P. Tang, and J. L. Ren, “An improved algorithm for Harris corner detection,” Opt. Precis. Eng., vol. 16, no. 10, pp. 1995–2001, 2008.

[34] W. E. Schaap, DTFE: The Delaunay Tessellation Field Estimator, Saarbrücken, Germany: Vdm Verlag Dr Müller, 2007.

[35] M. Greenspan and M. Yurick, “Approximate K-D tree search for efficient ICP,” in Proc. 4th Int. Conf. 3-D Digit. Imag. Modeling (DM), 2003, pp. 442–448.

[36] S. Du, N. Zheng, and S. Ying, “Affine iterative closest point algorithm for point set registration,” Pattern Recognit. Lett., vol. 31, no. 9, pp. 791–799, 2010.

[37] U. Qidwai and C. H. Chen, Digital Image Processing: An Algorithmic Approach With MATLAB, Boca Raton, FL, USA: CRC Press, 2017.

[38] S. Yadav, “Least squares best fit line method for the evaluation of measurement uncertainty with electromechanical transducers (EMT) with electrical outputs (EO),” MAPAN, vol. 25, no. 2, pp. 97–106, 2010.

[39] R. Adams and L. Bischof, “Seeded region growing,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 16, no. 6, pp. 641–647, Jun. 1994.

**YI XU (Student Member, IEEE)** received the B.S. degree in automation from Xi’an Jiaotong University, Xi’an, Shannxi, China, in 2017, where he is currently pursuing the master’s degree in control science and engineering.

His research interests include machine learning, artificial intelligence, model-based reinforcement learning, and their applications to intelligent systems.

**HAI-JUN RONG (Member, IEEE)** received the B.Eng. degree in precision instrument from Xi’an Technological University, China, in 2000, the M.Eng. degree in control theory and control engineering from Xi’an Jiaotong University, China, in 2003, and the Ph.D. degree in intelligent control from Nanyang Technological University, Singapore, in 2008.

From 2006 to 2008, she was a Research Associate and a Research Fellow with Nanyang Technological University. She has been an Associate Professor with the School of Aerospace, Xi’an Jiaotong University. Her research interests include neural networks, fuzzy systems, pattern recognition, and intelligent control. She is currently an Associate Editor of *Evolving Systems* (Springer).

**SHAOYI DU (Member, IEEE)** received the B.S. degrees in computational mathematics and computer science, the M.S. degree in applied mathematics, and the Ph.D. degree in pattern recognition and intelligence system from Xi’an Jiaotong University, China, in 2002, 2005, and 2009, respectively. From 2009 to 2011, he was a Postdoctoral Fellow with Xi’an Jiaotong University. From 2013 to 2014, he was with The University of North Carolina at Chapel Hill. He is currently a Professor with the Institute of Artificial Intelligence and Robotics, Xi’an Jiaotong University. His research interests include computer vision, machine learning, and pattern recognition.

**HONGMEI ZHANG** received the B.S., M.S., and M.D. degrees from Fourth Military Medical University, China, in 2000, 2003, and 2006, respectively. She is currently an Oncologist with Xijing Hospital. Her research interests include cancer treatment by targeted drug, DL to predict cancer treatment outcomes, drug toxicity, and drug-drug interactions.

---

**JING YANG** (Member, IEEE) received the B.S. and M.S. degrees in control science and engineering and the Ph.D. degree in pattern recognition and intelligent systems from Xi’an Jiaotong University, China, in 1999, 2002, and 2010, respectively.

From 1999 to 2003, she was a Research Assistant with the Institute of Automation, Xi’an Jiaotong University. Since 2003, she has been an Assistant Professor with the Department of Automation Science and Technology, Xi’an Jiaotong University. Since 2004, she has been a member of the Intelligent Vehicles Team, Institute of Artificial Intelligence and Robotics, Xi’an Jiaotong University. Her research interests include machine learning, reinforcement learning, and information theory and their applications to intelligent systems, such as autonomous vehicles.