PAPER

A class imbalanced wafer defect classification framework based on variational autoencoder generative adversarial network

IOP Publishing Measurement Science and Technology
Meas. Sci. Technol. **34** (2023) 024008 (14pp) https://doi.org/10.1088/1361-6501/ac9ed3

A class imbalanced wafer defect classification framework based on variational autoencoder generative adversarial network

Yitian Wang[1], **Yuxiang Wei**[1,2] **and Huan Wang**[2,*]

[1] Glasgow College, University of Electronic Science and Technology of China, Chengdu 611731, People's Republic of China
[2] Department of Industrial Engineering, Tsinghua University, Beijing, People's Republic of China

E-mail: huan-wan21@mails.tsinghua.edu.cn

Received 7 May 2022, revised 25 October 2022
Accepted for publication 31 October 2022
Published 18 November 2022

Abstract

Wafer defect classification (WDC) can be crucial to the wafer fabrication process. Engineers can quickly respond to improve the technological process, averting further defects through WDC. However, due to the complex fabrication steps, wafer defects are different in various types. This causes a severe data imbalance problem in WDC. To effectively solve the problem, this study introduces a class imbalanced WDC based on Variational Autoencoder Generative Adversarial Network (VAE-GAN). This framework consists of VAE-GAN and wafer defect classifier. Among them, VAE-GAN is responsible for creating new samples to solve the imbalance problem while the classifier is responsible for classifying wafer defect patterns. Specifically, VAE-GAN combines the advantage of a Variational Autoencoder (VAE) and generative adversarial network. VAE networks can produce subtle differences that do not affect the properties of the data when generating new images. At the same time, the proposed discriminator can help us constrain the generated images to be close to real samples and avoid irrational, feature-missing, and ambiguous samples. WM-811 K dataset is utilized to verify the above method. The experimental results validate that the samples generated by VAE-GAN have a significant improvement in the performance of the WDC system.

Keywords: wafer map, wafer defects classification, variational autoencoder, generative adversarial network

(Some figures may appear in colour only in the online journal)

1. Introduction

In the integrated circuits manufacturing industry, wafer fabrication is an essential procedure [1–3]. The manufacturing process of wafer is divided into two processes: front-end and back-end. Wafer fabrication, wafer probe test are included in the front-end process, and the back-end process refers to chip Packaging, testing and storage [4]. After the fabrication process, wafer tests are conducted to select the dies which meet production qualification [4–6]. Wafer defects are inevitable due to the complexity of wafer fabrication. Defect patterns need to be detected and classified, thereby reducing reliability issues and low yields of integrated circuits (ICs) [7]. Usually, engineers apply wafer defects patterns recognition system to determine the abnormal wafers. During the process, inline

* Author to whom any correspondence should be addressed.

metrology tools will be used to inspect wafer defects, and the wafer maps are created based on the defect locations. The collected wafer maps provide engineers with important defect information [8]. This information can help engineers quickly respond to the possible causes of wafer defects and adjust the process to reduce the yield of unqualified wafers [9]. Due to the ever-increasing demand for wafer yields in the semiconductor industry, the original engineers' visual inspection methods were not efficient enough to meet production demands [7]. Therefore, we need a system that automates wafer defect classification (WDC) tasks to replace manual labor and achieve higher production efficiency.

In recent years, convolutional neural network (CNN) have been applied to identify image patterns in many industries [10–12]. CNN is an end-to-end model that does not require manual extraction of image features. Through local perception and weight sharing, CNN can extract helpful features of images and continuously update the weights of convolutional kernels through the back propagation (BP) to obtain accurate identification results. Various CNN based methods have been conducted for the classification of wafer defects patterns in past years [12–14]. Li *et al* [8] propose a CNN based method for wafer map defect pattern classification and image retrieval. Kyeong *et al* [11] introduced a CNN model to classify mixed defect patterns within the framework of a separate classification model for each defect pattern and prosing a solution to the classification problem of mixed mix-type defect categories. Wei *et al* [7] proposed an MSF-Network to focus on the detailed features of the wafer map, which fully integrates the advantages of convolutional networks and transformers in detailed feature learning and global feature learning.

However, these methods do not consider the potential adverse effects of data imbalance in CNN-based WDC [15]. It is worth noting that insufficient data can lead to poor final classification accuracy in wafer classification problems. Because wafer fabrication is done in a highly automated and dust-free environment, the proportion of defective wafers is small relative to the normal ones. Since hundreds of complex processes are involved in wafer manufacturing, errors in different processes can lead to different types of wafer defects [7, 16]. Also, the occurrence of defects varies due to the differences between manufacturing processes. This results in an imbalance of defect categories in the wafer map samples.

To solve the problem of unbalanced data, scholars have utilized data processing methods such as oversampling and undersampling [17]. Nevertheless, these approaches can cause over-fitting problems in the model so that the model has high accuracy in training sets but generalizes poorly in test sets.

In recent years, scholars have begun to pay attention to the problem of data imbalance in wafer classification and deal with them with deep learning models. Wang *et al* [18] proposed a method for unbalanced classification of wafer defects based on a variational autoencoder augmented deep learning model. Tsai *et al* [13] proposed a similar lightweight depthwise convolutional autoencoder data augmentation model, which reduces model parameters and computation. Shen *et al* [5] proposed an augmentation method based on autoencoders after comparing different data generation approaches. Furthermore, an adaptive algorithm is utilized to determine the number of generated data volumes.

There are still some problems with these autoencoder-based data generation models, which are centered on the fact that the process of generating the data cannot be monitored. In the process of data augmentation by autoencoders, there are still several problems to be solved as follows: (a) **Irrationality**. Because the decoders of an autoencoder and variational autoencoder have poor robustness to noise. When decoding latent variables to generate images, sometimes it is possible to randomly generate wafer maps that do not conform to defect features or even irrational images due to the effect of adding noise. (b) **Missing Features**. Due to the randomness of the convolution downsampling process and the inexplicability of deep learning models, some information with significant physical meanings may be lost. Moreover, this information is likely to be part of a physically significant wafer's overall defect signature. This results in unclear defect features of the generated image [7]. (c) **Ambiguity**. A significant problem for autoencoders is that the decoded images are often blurry. Although we hope that the loss of the generated image and the original image to be small as possible, it is not easy to do in actual practicing. These blurred images can affect the classifier's judgment of image defect features.

To solve the above problems, it is necessary to build a feedback model in order to filter the irrational data, feature-missing data and ambiguous data. Motivated by this, this study introduces a class imbalanced wafer defect classification framework (CI-WDC) based on Variational Autoencoder Generative Adversarial Network (VAE-GAN). The framework consists of a data augmentation network and a wafer defect classifier. In particular, for data augmentation, we adopt VAE-GAN for image generation with higher efficiency and accuracy. Based on generating new data through a variational autoencoder, VAE-GAN introduced a discriminator to predict the reliability of generated data [19–21]. The discriminator has the ability to compare the distribution of image features to output the confidence that the image is true or false [22]. We use the original wafer map as our real image so that the discriminator can derive the validity of the image by comparing the distribution of the generated image and the original image [20, 22–24]. We can filter the generated data through a validation test to make the resulting data more rational and transparent.

There are indeed some other types of generative models that can be responsible for the generation of new wafer maps in imbalanced wafer classes, such as the original GAN [25], Energy-based GAN [26] and bidirectional GAN (BiGAN) [27]. However, these methods are aimed at solving complex real-world images, which is not so practical for the grey-scale data of wafer maps. Besides, these methods mentioned are mostly trained with random noise from the start. In our experiments, we found that it would produce many unexpected results and be really time-consuming. So, we would use VAE-GAN as our model, because it would better suit the task of generating simple grey-scale wafer maps.

The contributions of this paper are summarized as follows:

(a) This paper explores and studies the generation capabilities of the VAE network on defect wafer maps, proving it to be a powerful tool for wafer defect data augmentation.
(b) This paper proposes a feedback discriminator model based on the VAE network, which can monitor the generated data and improve the quality of the enhanced samples.
(c) This paper presents a novel framework (VAE-GAN based CI-WDC) for classifying class imbalanced defect wafer maps, perfecting the shortcomings of other autoencoder-based generative networks. This framework fully utilizes the generative ability of VAE and the discriminative ability of GAN. It obtains high-quality data augmentation results and high wafer classification accuracy.
(d) The framework proposed in this paper achieved a state-of-art performance on dataset WM-811 K. Furthermore, the visualization of augmented data also showed the ability of VAE-GAN to generate clear, diverse, and rational samples.

The following organization of the article is as follows: section 2 introduces WM-811 K dataset we used to validate data augmentation models; section 3 introduces the methods used and the overall structure of the framework; section 4 introduces the validation results, visualization results and comparison results of various methods on dataset WM-811 K; section 5 draws the conclusion.

2. Introduction to dataset

Wafer probing measures different electrical parameters on each device. The entire system of wafer prober and wafer test platform is called automatic test equipment. Each device is contacted by a set of probes that control the movement of the wafer. Resistance, voltage and current between wafer drain, gate and source can be measured by this device. After the measurement, we usually compare the measured test value with the specification limit of each parameter to judge whether the crystal equipment is defective. In the event of a parameter conflict, the device is discarded and the test is performed again for accurate values. The results obtained, including specification limits, x-y positions, wafer maps, etc are stored in a database as wafer data. In this experiment, we choose the WM-811 K dataset [3].

The WM-811 K dataset has 811 457 wafer maps collected from 46 393 batches in real manufacturing, of which domain experts labeled 172 950 wafers. The marked wafer defect classes are divided into nine types: Center, Donut, EdgeLocal, EdgeRing, Local, Scratch, None, Random, and NearFull. The Center has 4294 samples; Donut has 555 samples; EdgeLocal has 5189 samples; EdgeRing has 9680 samples; Local has 3593 samples; Random has 866 samples; Scratch has 1193 samples, and NearFull has 149 samples. And the none defect wafers are up to 147 431.

In the actual industrial production process shown in figure 1, the causes of these nine defect types cover most of the problems that may be encountered in wafer manufacturing. These nine categories are summed up by experts based on long-term

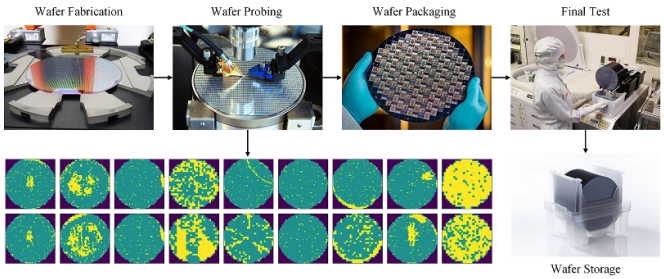

Figure 1. Wafer fabrication is the primary step, which produces the raw wafer disks. Wafer probing measures the electrical parameters on each one. Some important features will be measured by a set of devices. After the measurement, we compare the measured value with the specification limit to judge whether the wafer is defective. After sleeting the standardized wafer and storing the defect information, the wafers are packaged and sent to the final test. After all the tests, these disks are stored in specific containers.

industrial production experience, and we can consider such defect categories to be comprehensive enough. In an early paper discussing wafer defect class [12], and later papers considering wafer map defect recognition [28] which give detailed classification standards, both covered such type of defect classification method. And it is identical to the method we selected.

We need to perform data processing on the wafer maps labeled by experts to facilitate the subsequent training of deep learning networks. To ensure that the training data size is consistent, we use the nearest interpolation method to resize the wafer maps of different sizes to 64×64 while ensuring that the information loss of the wafer maps is minimized. After that, to classify the amorphous part, the intact part and the defective part more intuitively, we use One-hot encoding to convert the original grayscale images into RGB images. In the wafer maps, each pixel has three different possible values corresponding to three cases: when the pixel area is outside the wafer (Case 1); when the area is inside the wafer and normal (Case 2); when the area is inside the wafer and is defective (Case 3). We can use [1,0,0] to represent the pixel of Case 1, and [0,1,0] to represent the pixel of Case 2, [0,0,1] to represent the pixel point of Case 3. The entire encoding process follows the formula:

$$N(i,j,k) = 0, F(i,j) \neq k \tag{1}$$

$$N(i,j,k) = 1, F(i,j) = k \tag{2}$$

where $N(i,j,k)$ represents the value after encoding, and $F(i,j)$ represents the value before encoding [18].

3. Methodology

3.1. Overview of VAE-GAN framework

Since images generated by autoencoding will randomly produce irrationality, missing features and ambiguity, we need a new deep learning-based framework to solve these problems. This paper proposes a CI-WDC based on VAE-GAN.

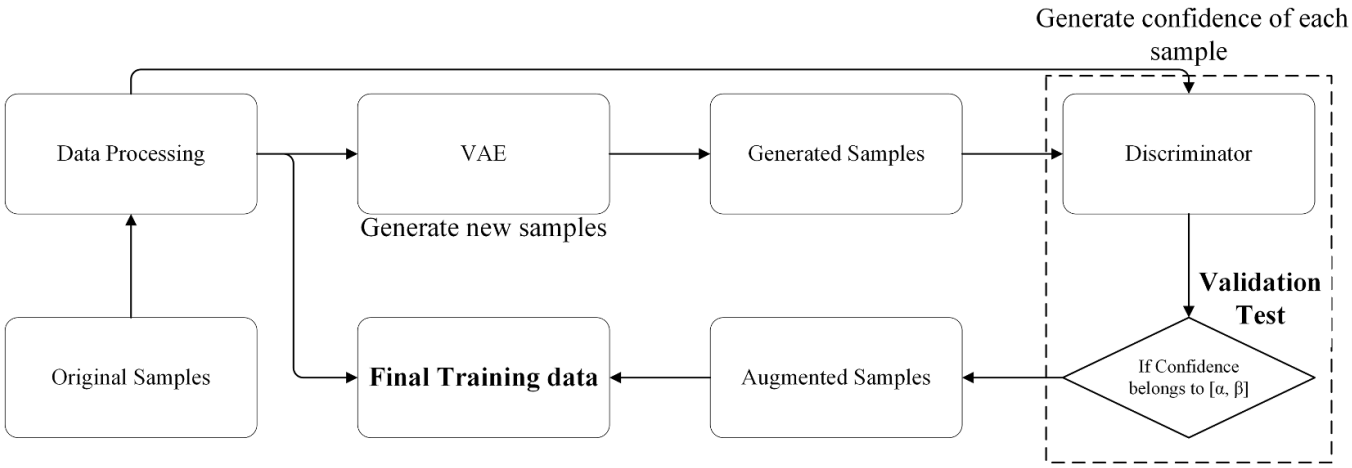

Figure 2. The overall flowcharts of VAE-GAN based data augmentation. The process is mainly divided into data processing; generative model training; sample generation and final CNN classification model training.

As shown in figure 2, this framework consists of VAE-GAN and a wafer defects classifier. The VAE-GAN consists of the encoder, decoder and discriminator. The encoder is used to obtain the original data's mean and variance vector using convolution and max-pooling layers. After constructing these two vectors to form a normal distribution, a latent variable z is sampled. While the decoders can decode the latent variable z into images using transposed-convolution and up-sampling. Before the latent variable is decoded, usually, we will add a small-scaled Gaussian noise to disturb the original distribution to create an image that is slightly different from the original image. In contrast, it still has the clear defect feature of wafers [22]. The discriminator can determine the confidence of the image by learning the feature distribution of the real image. It monitors the whole possess of generating new training data and filters out the equivocal and unqualified images to purify the dataset. The wafer defect classifier is then optimized with the augmented data.

3.2. Variational autoencoder

In this study, the basis for new wafer map generation is the VAE network. Define the wafer map samples as x; to generate a similar probability distribution of defected wafer map, the probability distribution $p(x)$ of x needed to be obtained [29]. Define the probability distribution $p(x)$ as follow:

$$p(x) = \sum_{z} p(x|z)P(z) \tag{3}$$

where z here is the latent variable from normal distribution, $p(x|z)$ describes a model that generates x from z, and $p(z)$ obeys the standard normal distribution $N(0, I)$. For each real sample $x^{(i)}$, define $p(z|x^{(i)})$ which obeys normal distribution to represent the unique latent distribution of each sample. Through multiple convolution layers with the kernels' size of 3×3, the distribution features of the wafer map defects are extracted and finally converted to a high dimensional vector of mean μ and variance σ^2. Mean μ expresses the average probability of a defective segment at a pixel, and variance σ^2 described fluctuations in this probability. We can sample the latent variables $z^{(i)}$ from the normal distribution constructed by μ and σ^2. This sampling process is implemented through the reparameterization trick. We define the parameters sampled from the normal distribution as ε. To sample $z^{(i)}$ from $p(z|x^{(i)}) \sim N(\mu^{(i)}, \sigma^{2(i)})$, firstly we sample ε from a standard normal distribution and then make $z^{(i)} = \mu^{(i)} + \varepsilon \times \sigma^{(i)}$. This is also a process of adding noise to the sampled latent variables $z^{(i)}$. $z^{(i)}$ is then constructed by the decoder into $\tilde{x}^{(i)}$. $\tilde{x}^{(i)}$ is the new wafer map we generate through the entire VAE network.

The decoding process can realize the construction of the original defect wafer maps through transposed convolution and restoring the original size of the wafer maps through upsampling. After $\tilde{x}^{(i)}$ is obtained, by minimizing the reconstruction loss between $\tilde{x}^{(i)}$ and $x^{(i)}$, the convolution kernel re-updates the weights and make the probability distribution of the reconstruction on $z^{(i)}$ more close to the original wafer maps. The reconstruction loss are defined as follow:

$$L_{\text{rec}} = -E_{q(z|x)}\left[log\, p(x|z;\theta)\right]. \tag{4}$$

By optimizing the generative model with such loss, we can achieve the constrain that $\tilde{x}^{(i)}$ is infinitely close to $x^{(i)}$.

Since $z^{(i)}$ is a variable sampled from the latent distribution $p(z|x^{(i)})$, it will be affected by the noise when reconstructing to generate $\tilde{x}^{(i)}$. This noise intensity is defined by the variance vector σ^2. When the decoder implements reconstruction of wafer maps, it will make σ^2 gradually tend to 0 to make the reconstruction loss smaller [29]. At this time, the model becomes a regular autoencoder that only output identical wafer defect features.

For the network to realize the variational ability instead of just autoencoding the original image, another loss function needs to be defined to prevent the noise from being

0. Since we have previously defined $p(z) \sim N(0, I)$, we can avoid the attenuation of noise intensity by constantly making $p(z|x^{(i)})$ approach $p(z)$ [29]. Therefore we need to introduce Kullback–Leibler divergence (KLD) to solve this problem. KLD measures the asymmetry of the difference between two probability distributions. When the KLD of two probability distribution drops, $p(z|x^{(i)})$ gets closer to $p(z)$.

$$L_{kl} = D_{kl}(p(z|x)||p(z)) \tag{5}$$

$$= \int_z p(z|x) \log\left(\frac{p(z)}{p(z|x)}\right) \mathrm{d}z. \tag{6}$$

When we continue to reduce KLD by neural networks, the noise intensity can gradually increase. This way, by decoding the noisy samples $z^{(i)}$, we get a new wafer map that differs in several pixels while keeping the general defect characteristics the same.

Therefore, the loss function of the entire variational autoencoder is defined as

$$L_{\text{vae}} = L_{\text{rec}} + L_{kl}. \tag{7}$$

We can generate the desired wafer maps by using this loss function to optimize the VAE model.

3.3. Variational autoencoder generative adversarial network

In an VAE network, the generated result is often difficult to be determined whether it is qualified or not through a formulaic decision algorithm [30]. The proposal of the discriminator model to the VAE network provides a new ideal to solve the problem: the feedback of the generative model is handed over to the discriminator to conduct the validation test. For the discriminator Dis, its original role is to distinguish whether the generated image is real or fake. It is a binary classifier trained by a multi-component BP network. The task performed by the discriminator can be regarded as a top-down process, which only needs to perform feature extraction on the input high-dimensional data to obtain low-dimensional discriminative information. It generates judgments on the confidence of wafer defect maps by directly understanding the difference between the original image and the generated image. Discriminator learns the feature of new wafer maps that are continuously updated by the VAE so that in each training batch, the ability to identify unqualified wafer maps can be improved [31]. Finally, after the VAE training is completed, the discriminator can obtain a highly accurate discriminative ability, making it possible to filter out random or abnormal results in the data generation process.

In our network, the loss function of the discriminator is defined as follows:

$$L_{bc} = -\frac{1}{\text{size}} \sum_{i=1}^{\text{size}} x_i \cdot \log \tilde{x}_i + (1 - x_i) \cdot \log(1 - \tilde{x}_i). \tag{8}$$

This is a binary cross-entropy loss function, where x_i represents the original image, $\tilde{x}_i$ represents the image generated by VAE, *size* refers to the overall sample numbers.

Algorithm 1: VAE-GAN FRAMEWORK

Input
Original Data $x^{(1)}, x^{(2)}, \ldots x^{(i)}$ trained by mini batch;
Output
Optimized Generator $G(\epsilon_1)$;
Optimized Discriminator $D(\epsilon_2)$;
Set $e = 0$ and each epoch E ;
Set early stop epoch E_{es};
While $e <= E$ **do**
  Encode $x^{(1)}, x^{(2)}, \ldots x^{(i)}$ and get the distribution of $p(z|x^{(i)}) \sim N(\mu^{(i)}, \sigma^{2(i)})$;
  Generate $z^{(1)}, z^{(2)}, \ldots z^{(i)}$ from sampling $p(z|x^{(i)})$: $z^{(i)} = \mu^{(i)} + \varepsilon \times \sigma^{(i)}$;
  Decode $z^{(1)}, z^{(2)}, \ldots z^{(i)}$ get $\tilde{x}^{(1)}, \tilde{x}^{(2)}, \ldots \tilde{x}^{(i)}$;
  Calculate L_{vae} and update ε_1 in G;
  IF $e <= E_{es}$
    Calculate L_{bc} and update ε_2 in D;
  IF an epoch is completed
    Set $e = e + 1$;

In VAE-GAN network, there are three major models, encoder, decoder and discriminator. The training of the VAE-GAN is the interactive training of the three models. The training steps of VAE-GAN are defined as Algorithm 1.

These steps are performed sequentially during each epoch. Usually, the ability of the discriminator varies in different training settings. To prevent the discriminator from discriminating the generated image as fake with a high level of confidence (usually lower than 0.3), we adopt the early stop strategy so that the discriminator stops optimizing after some epochs. Moreover, we will continue to train the generator to strengthen its generative ability.

3.4. Wafer maps generation

This study uses the earlier VAE-GAN model to perform data augmentation on the original wafer defect samples. The overall flow charts are shown in figure 2. Firstly, we performed data processing to get ideal sized wafer maps and used them for training our VAE-GAN models. After training the processed with multiple batches in VAE, we can get a well-trained encoder, decoder, and discriminator of these three models for the data generation process. The encoder can generate high-dimensional vectors of variance and mean from the original data and add noise to these two parameters to sample the latent variables of the wafer map features. The decoder can decode these latent variables into a generated image. The resulting image is usually similar to the original in overall defect distribution, with slight differences in subtleties. The discriminator can support the Validation Test to find the most suitable enhanced wafer map sample. After the loss converges, the models are frozen, and processed data are randomly put into the combined model of encoder and decoder. By collecting the output of the decoder, we get new wafer map samples.

3.5. Validation test

It is very important to judge the image's validity, which is related to the quality of the final augmented sample. In the discriminator, the high-dimensional image features are extracted as low-dimensional vectors and finally output as a probability value by the sigmoid function. To add up, As one of the most common activation functions in deep learning, sigmoid is usually widely used in binary classification and multi-label classification tasks. However, in Wasserstein generative adversarial network with gradient penalty (WGAN-GP) [32], the authors mentioned that using sigmoid as an activation function may lead to the problem of gradient explosion/vanishing in the training of the discriminator, which does have a negative impact on the training results of a model. But, since the target dataset of our proposed VAE-GAN is a wafer sample, which is grayscale data with a simple image structure, our model architecture has relatively few layers. Therefore, our model does not suffer from exploding/vanishing gradients during gradient propagation like the WGAN-GP model. Define probability as the confidence in the original data. Based on that, we propose a validation test, which aims to filter unqualified samples by the confidence of the generated images to find valid generated images. When the confidence of the image is high (up to 0.9), it is often too close to the original image and loses diversity. When the confidence of the image is low (lower than 0.6), it may accommodate irrational, feature-missing, or ambiguous samples, thereby polluting the enhanced data. We introduce a discriminant interval $[\alpha, \beta]$, $(0 < \alpha < \beta < 1)$. The generated wafer maps whose prediction result belongs to this interval are regarded as valid data, and the results that do not belong to this interval are filtered. When we expect more accurate data, we can increase the values of α and β and reduce the value of $\beta - \alpha$; when we expect more diverse data, we can decrease the values of α and β and increase the value of $\beta - \alpha$. Ultimately, we can adjust these values so that the final generated data is diverse and retains clear defect features. We can tune this discriminant interval by visualizing feedback on the augmented image and the accuracy of the classification model. This validation interval can make the distribution of our augmented wafer map not completely consistent with the original image, ensuring the diversity of the image; at the same time, it also ensures the accurate generation of wafer defects.

This validation test enables us to artificially control the characteristics of the enhanced samples. Compared with the unmonitored VAE generation results, we have more freedom to control the overall attributes of the enhanced samples. We can also generate data by adjusting the discriminant interval $[\alpha, \beta]$.

Also, this proposed method is a general method that can be transplanted into other GAN networks, thereby improving the selection ability of other GAN networks for generated samples. It should be noted that this method has different requirements for generators and discriminators compared to traditional GANs (usually more powerful generators are required), so when porting to other models, you may need to use early stop techniques to control training epochs.

3.6. CNN wafer defect classifier

In this framework, a CNN model is adopted as a classifier for wafer defect maps.

In the wafer defect classifier, we use the softmax activation function to achieve classification of multiple defect classes. The formula of softmax is expressed as follows:

$$\mathrm{softmax}(z_i) = \frac{e^{z_i}}{\sum_{n=1}^{N} e^{z_n}}. \tag{9}$$

Where z_i represents the output of the ith node, and N represents the number of output nodes. Through this function, we can convert the output values of multi-classification into probability distributions between 0 and 1. In order to better calculate the error between the predicted value and the true value, an appropriate loss function needs to be selected.

We take categorical-crossentropy loss as the loss function for the wafer defect map classifier.

$$L_{\mathrm{cc}} = -\sum_{i=1}^{\mathrm{Outputsize}} y_i \cdot \log \hat{y}_i \tag{10}$$

Before using the categorical–crossentropy loss function, the label of the categories used by the network are vectorized to multi-dimensional one-hot encoded vectors. The vectorized categories correspond to the multiple probability values trained by the neural network. Thus, the vector corresponding to the output with the largest probability value is the obtained label. Adam is used as the preferred optimizer for optimizing our loss function. Adam is based on the RMSprop algorithm and combines the advantages of the AdaGrad algorithm

$$E\left[g^2\right]_t = 0.9E\left[g^2\right]_{t-1} + 0.1g_t^2 \tag{11}$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{E[g^2]_t + \varepsilon}} g_t \tag{12}$$

where $E[g^2]$ is the mean of the sum of squared gradients. On the basis of this RMSprop, since the learning rate can be automatically adjusted in combination with Adagrad, the problem that the parameters are updated too slowly in the later stage is solved. We end up with a classifier with superior performance by continuously optimizing the categorical-crossentropy loss function until it converges.

4. Experimental validation

4.1. Experimental settings

In this section, we evaluate the augmentation results of the VAE-GAN framework on the dataset and the final classification accuracy with the WM-811 K dataset. First, we obtained a total of 25 507 samples other than 'none' type through data processing. Since there are far more samples (147 430) in the none category than in other defect categories, we selected

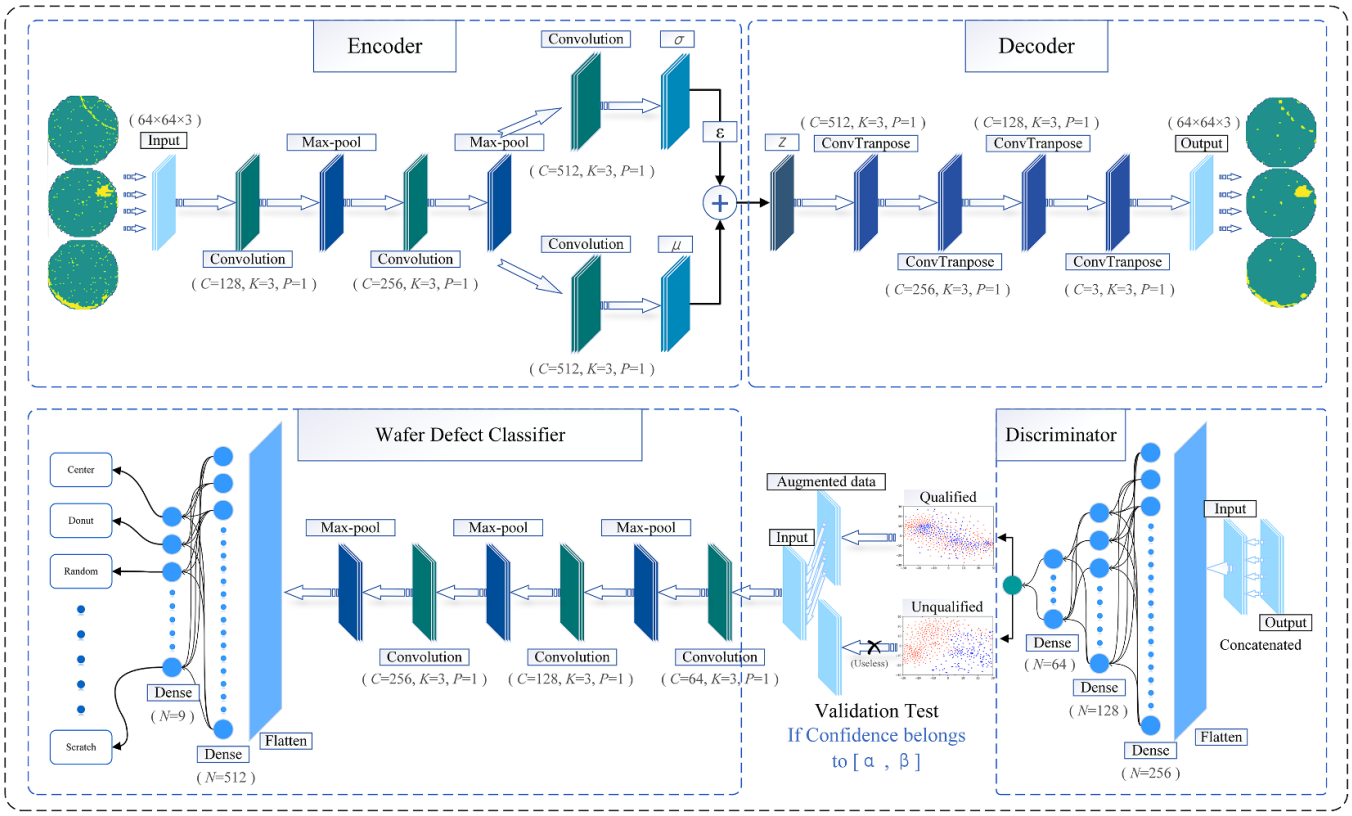

Figure 3. The overall model structure of VAE-GAN based CI-WDC framework. The framework is divided into four models, namely encoder, decoder, discriminator and wafer defects classifier. In convolution and transpose convolution layers, *C* denotes channels, *K* denotes kernel size, *P* denotes padding size. In Dense layers, *N* denotes the number of neurons.

5000 none defect samples for the training data. After that, 80% of the samples were randomly selected as the training set (24 406) and the rest as the test set (6101). This training is implemented under the acceleration of GTX2070s through Python3.8.5, Tensorflow2.2, and Keras2.3.1. The optimizer chosen in this experiment is Adam with learning rate 0.001 and $\beta_1 = 0.9$, $\beta_2 = 0.999$. The training batch size for VAE-GAN is 128, and 64 for CNN. From figure 3, we can get some hyperparameter settings for this model. For the Encoder, we use three layers of convolution, and the number of convolution kernels is 128, 256, and 512, respectively. The size of the convolution kernel is 3 times 3. For Decoder, we use four layers of deconvolution with 512, 256, 128, and 3 convolution kernels, respectively. For the discriminator, we used a multi-layer perceptron model with 256, 128, 64 and 1 neurons for each fully connected layer. For the convolution classifier, we used a three-layer convolution model with 64, 128, and 256 convolution kernels. After flattening, we used a fully connected layer to complete the classification task. The two fully connected layers had 512 and 9 neurons respectively. We have put the code on GitHub, please refer to the following link: https://github.com/wangyitian-charon/Wafer-defect-imbanlanced-problem-solving-by-VAE-GAN.

We compared the performance of different data augmentation models and defected class classification accuracy based on different WDC models. We used the convolution neural network wafer defect classification (CNNWDC) model as our baseline model, which means that all our data augmentation models will use this model as the wafer defect classifier.

The data augmentation methods used for comparison are as follows: Autoencoder-based WDC (AEWDC) [5] uses three layers of convolutional downsampling with channel sizes of 128, 256, 512 and kernel size of 3×3, and four layers of transposed convolution with channel sizes of 512, 256, 128, 3 and kernel size of 3×3. Also, we compared Variational Autoencoder-based WDC (VAEWDC) [18] which uses two layers of convolutional downsampling with channel sizes of 128 and 256, and adds two parallel convolutional layers with 512 channel sizes to obtain the vector of mean and variance with all of the convolution layers has the kernel sizes of 3×3. GAN-Based WDC (GANWDC) [33] uses a GAN-based adaptive balance generation network to generate simulated wafer defect maps with multiple dense layers applied in the generator and discriminator. In particular, we also tested the VAE-GAN model without adopting the Validation Test method and we maintain all training hyperparameters exactly the same as the VAE-GAN method. This model is abbreviated as VAE-GANWDC (NoVT, no validation test).

Also, we need to compare several other models that are not augmented by deep learning models. We used Oversampling-based WDC (OSWDC), where different classes are randomly replicated to address the imbalance. Also, we used

Table 1. Data augmentation statistics.

Defect Type	Original	After augmentation	Test
Center	3434	6434	858
Donut	443	3443	111
EdgeLocal	4150	7150	1037
EdgeRing	7743	10 743	1936
Local	2873	5873	718
Scratch	953	3953	245
None	4000	7000	1000
Random	692	3692	167
NearFull	118	3118	29
Total number	24 406	51 406	6101

the ResNet18 WDC model (ResNet18WDC) to make further comparisons.

We generated 3000 new data for each defect class and added them to the original training data.

To evaluate these models, we adopt four evaluation criteria: accuracy, recall, precision, FMeasure. These criteria are obtained through four outcome indicators, namely: true positive (TP); true negative (TN); false positive (FP); false negative (FN). The formula of each evaluation criteria is as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{13}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{14}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{15}$$

$$\text{FMeasure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \tag{16}$$

Among them, Accuracy is the proportion of accurate predictions in all predicted samples; Precision represents the proportion of TPs in the samples predicted as positive by the model; Recall evaluates the proportion of all actual positive cases in the predicted positive cases; the F1 measure is obtained after calculating the precision rate and recall rate, which is the harmonic average of the two, and it reflects the overall training result moderately.

4.2. Results of generation

The number of each category before and after data augmentation is shown in table 1. From the above table, it can be informed that in the original training data, the training categories of Donut, Scratch, Random, and NearFull are only 1.8%, 3.9%, 2.8% and 0.5%, respectively, compared to other training categories such as EdgeRing (31.7%) etc are seriously unbalanced. Through data augmentation, we successfully increased the original categories of Donut, Scratch, Random, and NearFull to 6.6%, 7.6%, 7.1% and 6.1% of the total augmented data. In addition to averaging the categories, we also expanded the total data volume from the original 24 406 samples to 51 406 samples.

Figure 4 demonstrates the generated wafer maps compared to the original wafer maps. From the graph, it is clear that the main features of each defect have been maintained, and on this basis, the diversity of samples is generated. In these images, we found no irrational data, and very few features of the images disappeared after augmentation while maintaining a high level of sharpness.

Figure 5 shows the t-distributed Stochastic Neighbor Embedding (t-SNE) dimensionality reduction plot of the generated and original samples. t-SNE is a machine learning algorithm used for dimensionality reduction, showing the closeness of data (or classification results) in low dimensions. From the figure, we can see that the red dots and the blue dots overlap each other. This demonstrates that the generated wafer samples have the same feature distribution as the original samples. It has been proved that the VAE network can create samples that ensure the same defect features.

4.3. Model evaluation

4.3.1. Results of classification. We use the test sets to evaluate the seven models separately and obtain the accuracy, recall, precision, and F-Measure values. These values can reflect the training performance and generalization ability of a model. Table 2 shows the experimental results of 4 criteria on each model. Table 3 shows the accuracy of each class on each model.

From table 2, the classification accuracy of VAE-GANWDC reached 96.82%, which is higher than the other 6 methods. The average accuracy is 96.42% for AEWDC, 96.22% for VAEWDC and 95.10% for GANWDC. For the model without deep learning data augmentation, we obtained an accuracy of 93.84% for OSWDC, 93.91% for ResNet18WDC and 93.82% for the CNNWDC model. Furthermore, our model outperformed its counterparts in precision, which was 96.80%, 1.38% higher than AEWDC (95.42%), 1.59% higher than VAEWDC (95.21%), 2.29% higher than OSWDC (94.51%), 2.37% higher than OSWDC (94.43%), 2.70% higher than ResNet18WDC (94.10%), 2.51% higher than GANWDC (94.29%). Also, the recall rate of VAE-GANWDC was 96.83%, higher than the other 6 models, which were 95.84%, 95.52%, 93.95%, 93.29%, 94.24%, 94.44% respectively. Lastly, the F-Measure of VAE-GANWDC is

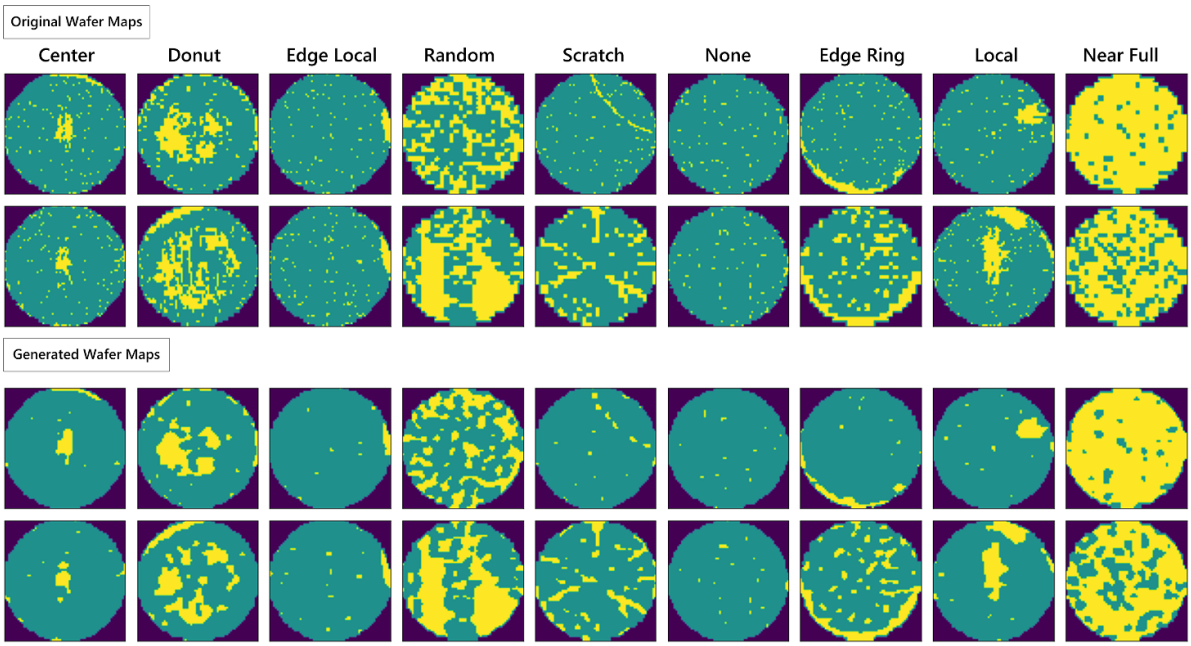

Figure 4. The figure demonstrates the comparison of Augmented data and Original data. From these two comparison charts, we can see that the images generated by the VAE-GAN model not only retain the original defect features but also have subtle but visible features with the original images. This proves that our generated images are of high quality.

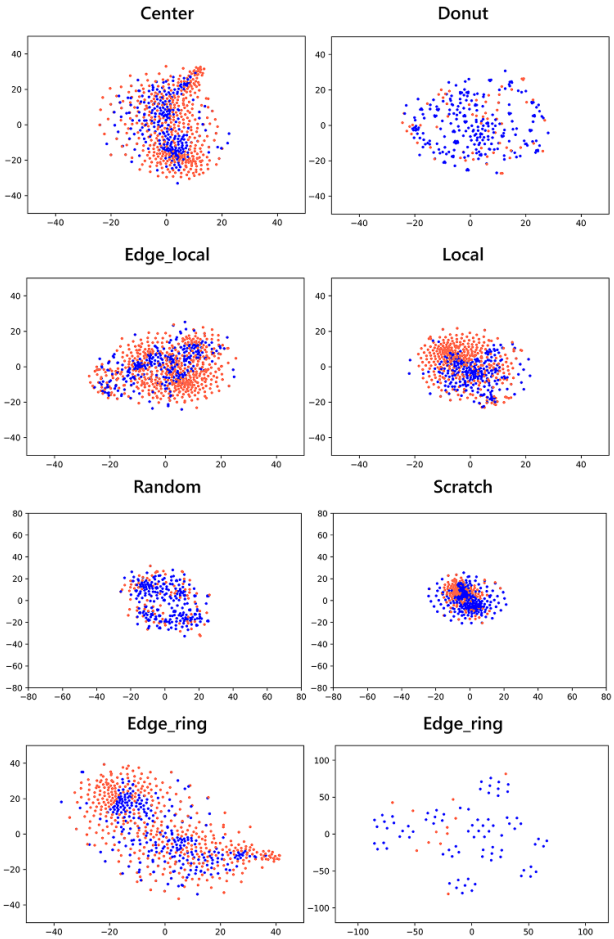

Figure 5. We present the generated and raw data with t-SNE dimensionality reduction, where blue dots represents the generated data and red dots represents the original data. From these images, we can see that our generated image has a very similar feature distribution to the original wafer image. This proves that our generated images can be used as feature maps of wafer defects for subsequent training.

Table 2. The experimental results of different models.

Model with augmentation	Accuracy	Recall	Precision	F-Measure
VAE-GANWDC	**0.9682**	**0.9680**	**0.9683**	**0.9682**
AEWDC	0.9642	0.9542	0.9584	0.9538
VAEWDC	0.9622	0.9521	0.9552	0.9609
GANWDC	0.9510	0.9429	0.9444	0.9451
VAE-GANWDC(NoVT)	0.9559	0.9541	0.9550	0.9544
Model without augmentation	Accuracy	Recall	Precision	F-Measure
OSWDC	0.9384	0.9451	0.9395	0.9412
CNNWDC	0.9382	0.9443	0.9329	0.9385
ResNet18WDC	0.9391	0.9410	0.9424	0.9399

Table 3. The experimental results of VAE-GANWDC, VAEWDC, AEWDC, OSWDC, CNNWDC, ResNet18WDC, GANWDC models on nine defect classes.

Model \ Defect Class	Center	Donut	Edge-Ring	Edge-local	Local	Random	Scratch	None	Near-Full
VAE-GANWDC	0.9793	0.9999	0.9831	0.9314	0.9081	0.9783	0.8942	0.9880	0.9999
AEWDC	0.9975	0.9861	0.9854	0.9641	0.9151	0.9751	0.8252	0.9854	0.9984
VAEWDC	0.9754	0.9584	0.9424	0.9241	0.9066	0.9775	0.8012	0.9903	0.9809
OSWDC	0.9450	0.9824	0.9945	0.9451	0.9217	0.9015	0.8521	0.9123	0.9670
CNNWDC	0.9790	0.9912	0.9961	0.8752	0.7631	0.9671	0.8451	0.9901	0.9999
ResNet18WDC	0.9898	0.9954	0.9412	0.9541	0.9100	0.9546	0.7666	0.9530	0.9998
GANWDC	0.9999	0.9984	0.9945	0.9315	0.9012	0.9577	0.8054	0.9815	0.9922
VAE-GANWDC(NoVT)	0.9794	0.8244	0.9935	0.8574	0.8145	0.8964	0.8215	0.9852	0.9234

96.82%, outperforms others which are respectively 95.38%, 96.09%, 94.12%, 93.85%, 93.99%, 94.51%.

We can see from the above results that the model's accuracy without data augmentation is generally lower than that of the model with data augmentation. This represents that data augmentation does have a considerable improvement in the classification accuracy of the CNNWDC model. Among them, VAE-GANWDC has the best performance under different evaluation criteria, which shows the superior comprehensiveness and sensitivity of the model for different defect feature data classification. At the same time, we also added a set of ablation experiments on the impact of the Validation Test on the original model. We can find that the full VAE-GAN performs better than the VAE-GAN without the Validation Test. This means that our proposed method works well for selecting generated data.

Table 3 shows the classification accuracy of each defect class on different models. For VAE-GANWDC, the accuracy of each defect class has reached more than 90%. Also, it can be seen from the figure that VAE-GANWDC has almost 100% accuracy in the identification of Donut and NearFull. For the category of Center defects, the recognition accuracy of VAE-GANWDC is moderately high among all models (97.3%). For the defect category, EdgeRing, EdgeLocal, Random, VAE-GAN also achieved excellent performance (98.31%, 93.14%, 97.83%) among all models. For none defect wafers, the recognition accuracy reaches 98.80%, which is comparable to the accuracy of most models. In particular, VAE-GAN has outstanding performance in classifying scratch defects. VAE-GAN has an accuracy of 89.42%, while most of the other models have a recognition accuracy between 85% and 80%.

The VAE-GANWDC framework has a more vital recognition ability for categories with a low classification accuracy than other models, such as the scratch. The VAE-GAN model can generate diverse and clear featured training data, which better helps CNN learn the feature distribution of specific defect categories, thereby achieving robustness to different wafer defect maps.

4.3.2. Confusion matrix (CM). In model evaluation, we often need to know the training accuracy of each class so as to evaluate the model more completely. The accuracy of the classification results can be displayed in a CM. A CM is a specific two-dimensional matrix. Its columns represent the predicted categories; rows represent actual categories. The values on the diagonal of this matrix represent the percentage of correct predictions; the off-diagonal elements are the percentages of predictions to other classes. Figures 6–8 demonstrate the CM of the CNNWDC model and VAE-GANWDC model without the validation test and the VAE-GANWDC model we finally proposed. From the three matrices, we can see a significant improvement in classification accuracy after data augmentation, and also more improvements after we activated validation test in VAE-GAN model.

This also demonstrates the advantage of our model in the high recognition accuracy of each class. We have greatly improved the classification accuracy of weak categories to ensure that the accuracy rates of other categories are higher

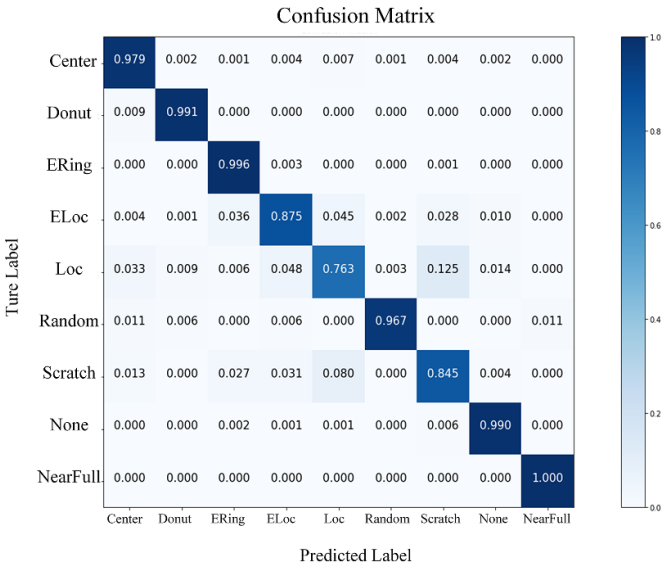

Figure 6. Confusion Matrix of CNNWDC. In this confusion matrix diagram, we can find that the classification accuracy of the model is not ideal when classifying Edge Local, Local and Scratch categories, and samples are often misclassified.

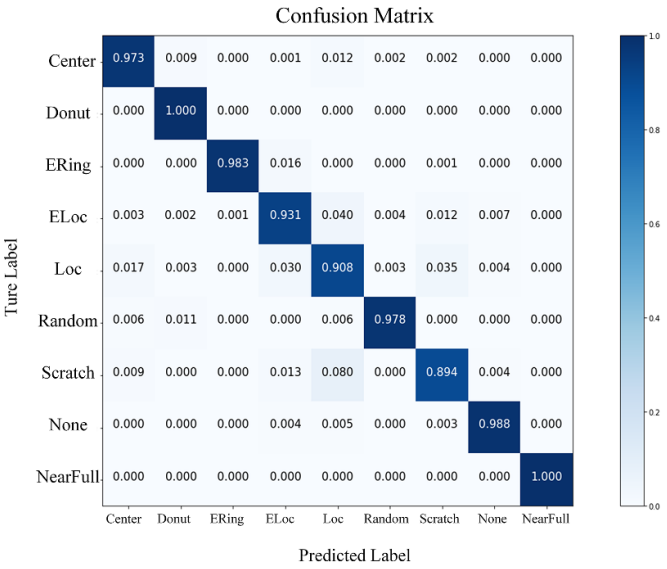

Figure 8. Confusion Matrix of VAE-GANWDC. In this confusion matrix, we can find that the original low-accuracy categories have been greatly improved, especially the scratch category has an accuracy of nearly 90%. This is a great improvement over other models.

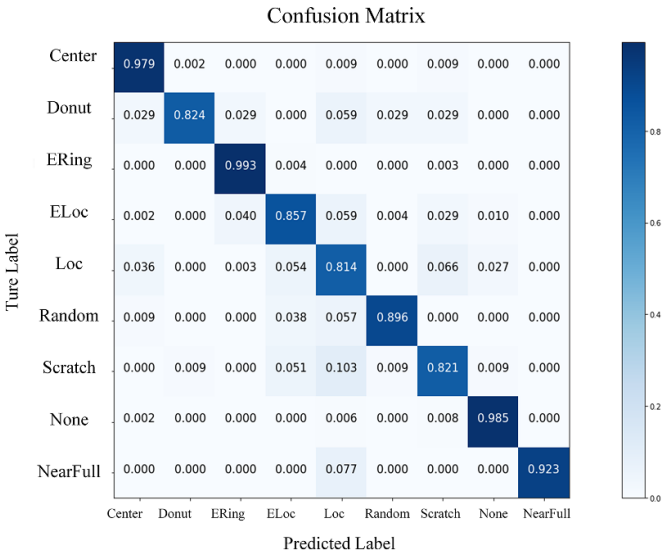

Figure 7. Confusion Matrix of VAE-GANWDC(NoVT). In confusion matrix diagram, we can find that although the classification accuracy of the Local category is improved, the accuracy of other categories fluctuates slightly, and the overall accuracy is not much improved compared to the CNN model.

than average, thereby achieving a higher overall accuracy rate. From the figure, we can see that the accuracy of almost every category has improved after data augmentation. The most obvious ones are the Local and Scratch defect categories, which have increased from 76.31% and 90.8% to 84.51% and 89.42%, respectively.

Additionally, From the CM of CNNWDC (figure 6), we can see that 8.00% of Scratch images are recognized as Local, and 12.5% of Local images are recognized as Scratch; 8.0%, 3.1% of Local defect wafers are recognized as Scratch and Edge-Local; 4.5%, 3.6% of EdgeLocal defect wafers are recognized as Local and EdgeLocal. The CM of VAE-GANWDC(NoVT) (figure 7) shows that 10.0% of Scratch images are recognized as Local and 5.1% are recognized as Edge Local; 6.6%, 5.4% of Local defect wafers are recognized as Scratch and EdgeLocal; 5.9% of EdgeLocal defect wafers are recognized as Local. The CM of VAE-GANWDC (figure 8) shows that 3.50% of Scratch images are recognized as Local; 8.0%, 1.3% of Local defect wafers are recognized as Scratch and EdgeLocal; 4.0% of EdgeLocal defect wafers are recognized as Local. In this regard, we believe that the augmented training set solves the generalization defect caused by the imbalanced data of the original data set to a certain extent. As a result, the accuracy of the above categories with obvious identification errors is increased.

4.3.3. t-SNE visualization of predict results. To better demonstrate the classification accuracy of our baseline model CNNWDC and our proposed VAE-GANWDC, we use t-SNE to display the dimensionality reduction visualization of the prediction results [34]. The defect category represented by red is Center, the defect category represented by tomato color is Donut, orange represents EdgeRing, cyan represents EdgeLocal, black represents Local, light coral represents Random, yellow represents Scratch, blue represents None, and grey represents NearFull. In figure 9(a), the yellow, black, and light coral colors overlap in many areas. This represents the flawed ability of the simple CNN model to recognize Scratch, Local and Random defects. In figure 9(b), most points are scattered with different color groups. These scattered color circles hardly contain points of other colors, indicating that our training successfully separated different classes of defects. It is evident that the

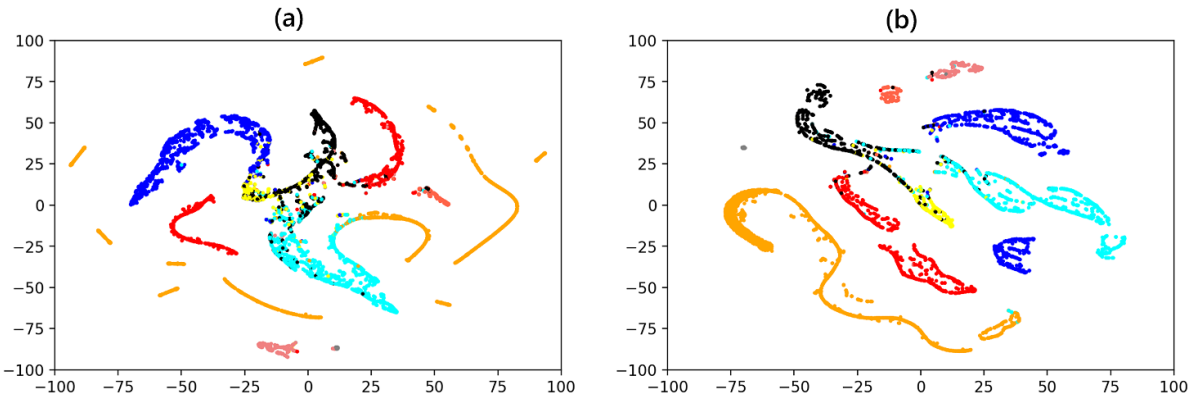

Figure 9. t-SNE comparison before and after VAE-GAN processing. The colors correspond to different types of defect. The relationships are follows: red: Center, tomato: Donut, orange: EdgeRing, cyan: EdgeLocal, black: Local, gray: NearFull, lightcoral: Random, yellow: Scratch, and blue: None. (a) t-SNE of CNNWDC. (b) t-SNE of VAE-GANWDC.

accuracy of the CNN after training on the augmented data has improved.

5. Discussions

5.1. Discussions of contributions

Based on our literature review (See in the Introduction), we found that most of the current papers' solutions to wafer defect imbalance datasets are relatively simple autoencoder-based algorithms. There are still some problems with these autoencoder-based data generation models, which are centered on the fact that the process of generating the data cannot be monitored. In the process of data augmentation by autoencoders, there are still several problems to be solved such as Irrationality, Missing Features and Ambiguity. Still, there are generative architectures that have been developed other than autoencoders, such as WGAN-GP, energy-based GAN, BiGAN and more. However, the wafer maps are small and simple grayscale data and the algorithms mentioned above are based on natural images with complex backgrounds, which is not very practical for accomplishing our target tasks. Therefore, inspired by these algorithms, we integrated some self-supervised generative models, combined the advantages of variational autoencoders and generative adversarial networks and proposed VAE-GAN.

Our experiments on the real wafer defect data s*et al*so prove that our generated data is very similar to the real wafer data. From figure 4 of the article, we can see that our generated images are extremely similar to the original images, but still slightly different, which represents the good quality of our generation images at the same time, the t-SNE dimensionality reduction plot in figure 5 also shows that the distribution of our generated data is very similar to the original data. This is a good verification of the quality of our generated data. Lastly, by supplementing these generated data into the training set, we also found that the performance of the model has also been well improved.

5.2. Limitations and future improvements

During our experiments, we found that the VAE-GAN model has several shortcomings. First of all, the two hyperparameters in the Validation Test we proposed are determined by human prior knowledge, which is often difficult to be very precise, and the results are not fully optimized. At the same time, since VAE-GAN is a two-stage method, it includes two processes image generation and image recognition. Therefore, this method lacks real-time performance. In the future, we will apply this method to other data sets to solve specific engineering problems; at the same time, we will also use advanced algorithms to achieve automatic tuning of hyperparameters in the original model; finally, by optimizing the original algorithm and model architecture, we can improve the real-time performance of our model.

6. Conclusion

In this paper, we present a novel framework, the VAE-GAN-based CI-WDC. and complete the augmentation of the raw wafer data, increasing the variety of generated samples while maintaining the quality of the generation. The VAE-GAN model is a generative adversarial model based on the VAE network. In particular, the VAE network is responsible for generating new samples. The introduced discriminator can identify unqualified samples by training the generated and original samples. When we perform data augmentation, the images generated by VAE will go through the discriminator's validation test, the samples that pass the test are responsible for CNN training.

Our experiments fully demonstrate the excellent results of the framework on the class imbalanced wafer dataset and achieve the best performance in comparing multiple models. We demonstrate the improvement of VAE-GAN data augmentation over simple CNN models through different evaluation criteria, confusion matrices, and t-SNE dimensionality reduction visualizations. Combining all the above results, the

augmentation ability of the VAE-GAN model for imbalanced data is evident.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://www.kaggle.com/datasets/qingyi/wm811k-wafer-map.

Acknowledgment

This work was supported by Innovation Fund of Glasgow College, University of Electronic Science and Technology of China.

ORCID iD

Huan Wang https://orcid.org/0000-0002-1403-5314

References

[1] Kary Chen W-T. and Huang C H J 2002 Practical "building-in reliability" approaches for semiconductor manufacturing *IEEE Trans. Reliab.* **51** 469–81
[2] Nag S, Makwana D, TejaR S C, Mittal S and Mohan C 2022 Wafersegclassnet—a light-weight network for classification and segmentation of semiconductor wafer defects *Comput. Ind.* **142** 103720
[3] Schrunner S, Jenul A, Scheiber M, Zernig A, Kaestner A and Kern R 2019 A health factor for process patterns enhancing semiconductor manufacturing by pattern recognition in analog wafermaps *2019 IEEE Int. Conf. on Systems, Man and Cybernetics (SMC)* pp 3555–60
[4] Mat Jizat J A, Abdul Majeed A P P, Nasir A F A, Taha Z and Yuen E 2021 Evaluation of the machine learning classifier in wafer defects classification *ICT Express* **7** 535–9
[5] Shen P-C. and Lee C-Y. 2022 Wafer bin map recognition with autoencoder-based data augmentation in semiconductor assembly process *IEEE Trans. Semicond. Manuf.* **35** 198–209
[6] Chauhan K K, Joshi G, Kaur M and Vig R 2022 Semiconductor wafer defect classification using convolution neural network: a binary case *IOP Conf. Ser.: Mater. Sci. Eng.* **1225** 012060
[7] Wei Y and Wang H 2022 Mixed-type wafer defect recognition with multi-scale information fusion transformer *IEEE Trans. Semicond. Manuf.* **35** 341–52
[8] Li W, Zhong X, Shao H, Cai B and Yang X 2022 Multi-mode data augmentation and fault diagnosis of rotating machinery using modified acgan designed with new framework *Adv. Eng. Inform.* **52** 101552
[9] Li K S-M, Chen L L-Y, Cheng K C-C, Liao P Y-Y, Wang S-J, Huang A Y-A, Chou L, Tsai N C-Y and Lee C-S 2022 Testdna-e: wafer defect signature for pattern recognition by ensemble learning *IEEE Trans. Semicond. Manuf.* **35** 372–4
[10] Batool U, Shapiai M I, Tahir M, Ismail Z H, Zakaria N J and Elfakharany A 2021 A systematic review of deep learning for silicon wafer defect recognition *IEEE Access* **9** 116572–93
[11] Kyeong K and Kim H 2018 Classification of mixed-type defect patterns in wafer bin maps using convolutional neural networks *IEEE Trans. Semicond. Manuf.* **31** 395–402
[12] Phua C and Theng L B 2020 Semiconductor wafer surface: automatic defect classification with deep cnn *2020 IEEE Region 10 Conf. (TENCON)* pp 714–9
[13] Tsai T H and Lee Y C 2020 A light-weight neural network for wafer map classification based on data augmentation *IEEE Trans. Semicond. Manuf.* **33** 663–72
[14] Wang S, Yan S, Shen Q, Luo C, Ai J, Li L, Wang D, Ding S and Xia Q 2021 Wafer defect map similarity search using deep learning in semiconductor manufacturing *2021 China Semiconductor Technology Int. Conf. (CSTIC)* pp 1–4
[15] Yu N, Chen H, Xu Q, Hasan M M and Sie O 2022 Wafer map defect patterns classification based on a lightweight network and data augmentation *CAAI Trans. on Intelligence Technology* (https://doi.org/10.1049/cit2.12126)
[16] Kim Y, Cho D and Lee J-H 2020 Wafer map classifier using deep learning for detecting out-of-distribution failure patterns *2020 IEEE Int. Symp. on the Physical and Failure Analysis of Integrated Circuits (IPFA)* pp 1–5
[17] Liu C, Jin S, Wang D, Luo Z, Yu J, Zhou B and Yang C 2022 Constrained oversampling: an oversampling approach to reduce noise generation in imbalanced datasets with class overlapping *IEEE Access* **10** 91452–65
[18] Wang S, Zhong Z, Zhao Y and Zuo L 2021 A variational autoencoder enhanced deep learning model for wafer defect imbalanced classification *IEEE Trans. Compon. Packag. Manuf. Technol.* **11** 2055–60
[19] Tang Y, Zhao Z, Ye X, Zheng S and Wang L 2020 Jamming recognition based on ac-vaegan *2020 15th IEEE Int. Conf. on Signal Processing (ICSP)* vol 1 pp 312–5
[20] Krajewski R, Moers T and Eckstein L 2019 Vegan: Using gans for augmentation in latent space to improve the semantic segmentation of vehicles in images from an aerial perspective *2019 IEEE Winter Conf. on Applications of Computer Vision (WACV)* pp 1440–8
[21] Radford A, Metz L and Chintala S 2016 Unsupervised representation learning with deep convolutional generative adversarial networks *CoRR* (arXiv:1511.06434)
[22] Altay F and Velipasalar S 2018 Image completion with discriminator guided context encoder *2018 52nd Asilomar Conf. on Signals, Systems and Computers* pp 2220–4
[23] Peng Y, Pan X, Wang S, Wang C, Wang J and Wu J 2021 An aero-engine rul prediction method based on vae-gan *2021 IEEE 24th Int. Conf. on Computer Supported Cooperative Work in Design (CSCWD)* pp 953–7
[24] Hou X, Shen L, Sun K and Qiu G 2017 Deep feature consistent variational autoencoder *2017 IEEE Winter Conf. on Applications of Computer Vision (WACV)* pp 1133–41
[25] Larsen A B L, Sønderby S K, Larochelle H and Winther O 2016 Autoencoding beyond pixels using a learned similarity metric *Proc. of The 33rd Int. Conf. on Machine Learning (Proc. of Machine Learning Research)* vol 48, ed M F Balcan and K Q Weinberger (New York: PMLR) pp 1558–66
[26] Zhao J J, Mathieu M and LeCun Y 2016 Energy-based generative adversarial network (arXiv:1609.03126)
[27] Donahue J, Krähenbühl P and Darrell T 2017 Adversarial feature learning (arXiv:1605.09782)
[28] Maksim K, Kirill B, Eduard Z, Nikita G, Aleksandr B, Arina L, Vladislav S, Daniil M and Nikolay K 2019 Classification of wafer maps defect based on deep learning methods with small amount of data *2019 Int. Conf. on Engineering and Telecommunication (EnT 2019)* (Institute of Electrical and Electronics Engineers Inc.)
[29] Kingma D P and Welling M 2014 Auto-encoding variational bayes *CoRR* (arXiv:1312.6114)
[30] Col P and Ertekin S 2021 Feature dimensionality reduction with variational autoencoders in deep bayesian active

learning *2021 29th Signal Processing and Communications Applications Conf. (SIU)* pp 1–4
[31] Mescheder L, Nowozin S and Geiger A 2017 Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks *Int. Conference on Machine Learning (PMLR)* pp 2391–400
[32] Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V and Courville A C 2017 Improved training of wasserstein gans *Advances in Neural Information Processing Systems 30 (NIPS 2017)*
[33] Wang J, Yang Z, Zhang J, Zhang Q and Chien W-T K 2019 Adabalgan: an improved generative adversarial network with imbalanced learning for wafer defective pattern recognition *IEEE Trans. Semicond. Manuf.* **32** 310–9
[34] van der Maaten L and Hinton G 2008 Visualizing data using t-SNE *J. Mach. Learn. Res.* **9** 2579–605