

Anomaly detection in random circuit patterns using autoencoder

Hiroshi Fukuda^{✉*} and Tsuyoshi Kondo

Hitachi High-Tech Corporation, Tokyo, Japan

Abstract

Background: Systematic/stochastic pattern defects affect the production yield of integrated circuits (IC) containing trillions of 10-nm level features.

Aim: We detect pattern anomalies/defects from images obtained from scanning electron microscopes (SEM) for random/arbitrary IC patterns without using design data.

Approach: We decompose SEM images into small sub-images and apply an identical autoencoder to each of them to detect anomalies. The astronomical varieties in random IC patterns are reduced into limited varieties in elementary patterns, which are coded onto limited dimension latent vectors in autoencoder. The discrepancy between autoencoder input and output represents a deviation of local pattern shapes from ideal or allowed ones and is used as an index of anomaly.

Results: A wide variety of anomalies/defects are detected in regular and random IC patterns fabricated by extreme ultra-violet lithography without prior knowledge about anomalies at a high signal-to-noise ratio within a time shorter than the typical image acquisition time of SEMs. They include missing/necking in holes/trenches, collapsing/breaking in lines, various local pattern distortion/deformation, and tiny particles. Frequency and spatial distributions of the discrepancy index are sensitive to process changes and can be used for visualizing the sign or causes of anomalies.

Conclusions: The method is effective for inspecting memory and random logic ICs with high-speed SEMs.

© The Authors. Published by SPIE under a Creative Commons Attribution 4.0 International License. Distribution or reproduction of this work in whole or in part requires full attribution of the original publication, including its DOI. [DOI: [10.1117/1.JMM.20.4.044001](https://doi.org/10.1117/1.JMM.20.4.044001)]

Keywords: integrated circuits; defect inspection; anomaly detection; machine learning; autoencoder; lithography.

Paper 21054 received Jun. 10, 2021; accepted for publication Oct. 20, 2021; published online Nov. 5, 2021.

1 Introduction

The densities of integrated circuits (IC) are increasing with continuously shrinking circuit patterns, which is achieved by the introduction of EUV lithography and/or multiple patterning technologies. As a result, more than a trillion features as small as 10 nm are formed on a 300-mm wafer or even on an exposure field, and this imposes another challenge to guarantee defect-free for these circuit patterns. In particular, the following two types of defects are becoming major concerns; systematic defects are generated at particular spots in design patterns where the process windows are narrow (aka hot-spots or weak-spots).¹ Stochastic defects are generated randomly in EUV resist patterns due to photon shot noise and discrete/probabilistic natures of materials, and their probability exponentially increases with decreasing feature size.² The sizes of both types of defects are typically smaller than minimum circuit feature size and they are highly sensitive to process conditions, thus are generated by unexpected variations in process conditions.

Conventionally, the inspection of these defects has been carried out by optical inspection tools.³ Since it is not clear if the optical tools can keep enough sensitivity for ever-shrinking

*Address all correspondence to Hiroshi Fukuda, hiroshi.fukuda.zp@hitachi-hightech.com

defects size, high-speed SEM inspection including multi-electron-beam systems are also expected.⁴ From inspected images, the pattern anomalies/defects are usually detected based on comparison with some reference data, as in die-to-die comparison in optical tools. A high-resolution capability of SEMs enables direct comparison of circuit patterns between inspected images and design layout data, but this generally requires access to a huge amount of design layout data particularly in random logic IC.⁵

In the area of anomaly detection, machine learning (ML) has widely been adopted.⁶ In general, there are two approaches in applying ML to anomaly detection; in one approach, machines are trained with abnormal data in advance, then they find abnormal parts of the data to inspect. In the other approach, the machines are trained with normal data in advance, and then, they find the parts that do not match the learned normal data from the data to inspect.⁷

In applying the former approach to detecting IC pattern anomaly/defects from SEM inspected images,⁸ it is difficult to obtain sufficient numbers of training data, because anomaly data required for training are extremely rare in general. Additionally, tiny defects/anomalies need to be filtered from the normal pattern background, which has extremely wide varieties. In contrast with the latter approach, we can easily obtain a number of normal pattern data as training data, although the problem with the huge pattern variations remains. Several attempts have been made to apply ML to evaluating IC patterns/masks using normal data.^{9,10} This study discusses the potential of the second approach for direct inspection and analysis of images taken by SEM inspection tools.

In what follows, we introduce our approach based on autoencoders in Sec. 2. Its characteristics and application results are discussed in Secs. 3 and 4. All the samples and SEM images used in this study were prepared and obtained in IMEC.

2 Method

2.1 Basic Concept

An autoencoder is a neural network that has widely been used in various applications including anomaly detections.^{6,7} It consists of an encoder and decoder. The encoder compresses the input data into the middle layer called a latent vector, and the decoder decompresses the data from the latent vector to generate a representation as close to the original input as possible. The latent vector has a smaller dimension than the input data and represents a data-specific and lossy version of the trained data. The network is learned to minimize the reconstruction loss, a distance function, or the amount of information loss between the compressed and the decompressed representations. After training, the autoencoder reconstructs input data only when input data is similar to those used for training data. Thus, by training the autoencoder with normal data, it outputs normal data only when the input is normal, and this characteristic of autoencoders can be used for detecting anomalies. Here, we investigate the application of autoencoders to inspecting IC patterns. In this section, we introduce our basic approach.

Practically, several problems arise when applying the autoencoder to detecting defects/anomalies in IC patterns from images obtained by SEM. First, the amount of data to inspect is huge (1-mm square area contains 10^{12} of 1-nm square pixels). Second, the variation in circuit patterns contained in these areas is astronomical in particular for random logic IC. Third, defect signals [and signal-to-noise ratio (SNR)] are low due to their tiny sizes (<10 nm) compared with circuit patterns (composed of 10- to 20-nm width lines in a 0.5- to 2- μm square field of view (FOV) of SEM, for example).

Our approach to these problems is as follows (Fig. 1): We decompose IC pattern images into a number of small sub-images using a clipping window and apply an identical autoencoder to each of those sub-images to detect anomalies. The astronomical number of varieties in complex random patterns are reduced into a limited variation in elementary patterns within clipped areas, which can be coded onto limited dimension latent vectors and reconstructed in autoencoder. The discrepancy between input and output of autoencoder represents any deviation of local pattern shape from ideal, expected, or allowed ones and is used as an index of anomaly. By decomposing original images into small sub-images, we limit the size of the autoencoder and decrease the

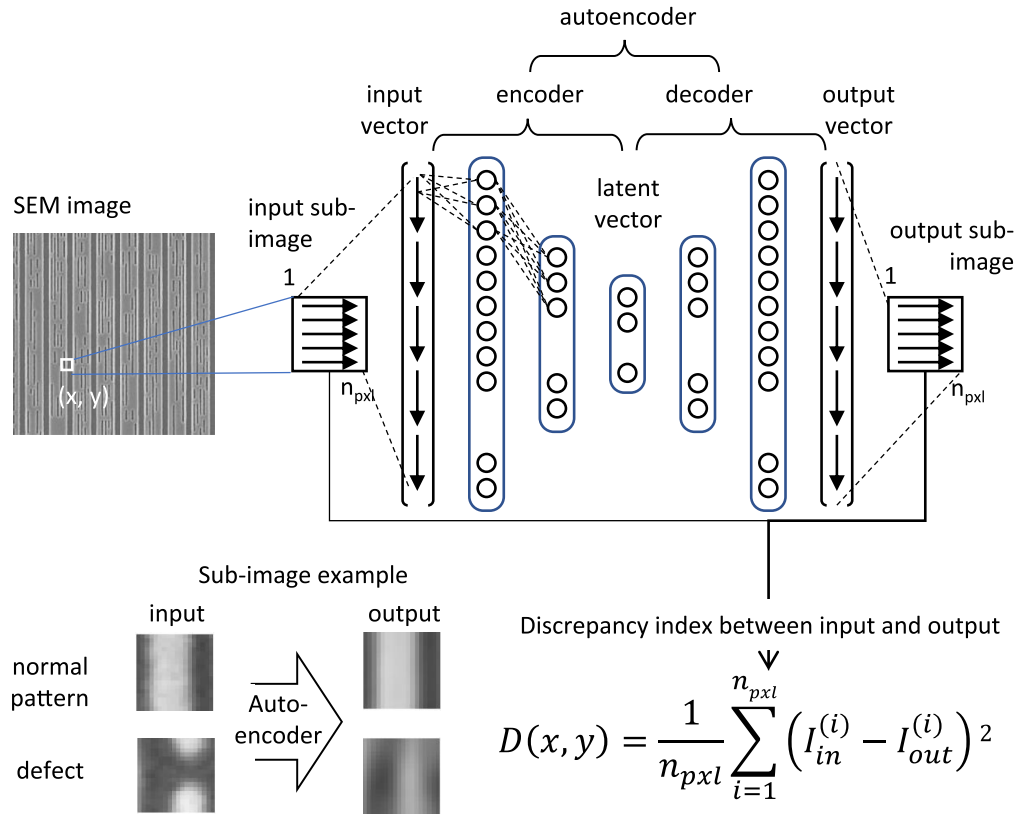


Fig. 1 Pattern anomaly detection concept based on autoencoding of decomposed IC pattern images.

pattern variation in sub-images within practical (trainable) range while enhancing defect SNR. This approach can be applied to arbitrary complex pattern features as in random logic IC.

For a simple example, suppose that we clip F square sub-area (F: minimum line width) from random circuit patterns designed under the so-called Manhattan layout rule Fig. (2, left). The possible pattern variations within the clipped area are limited to patterns as illustrated in the right of Fig. 2. Reversely, an arbitrarily designed layout can be expressed by properly combining sub-patterns in this reduced elementary pattern set. Thus, by training the autoencoder to reconstruct patterns contained in the elementary patterns set in Fig. 2 only, in principle, we can detect anomalies in arbitrary patterns designed as a combination of them.

For real pattern features fabricated by projecting (exposing) properly designed mask, the distances between neighboring two edges and radius of curvature of edges do not exceed certain values. Any inter-edge distance or radius of curvature that is smaller than this criterion is detected as an anomaly. Thus, we train an autoencoder/network with sub-images for as wide variations as possible, which can appear on normally exposed wafers. It is worth noting that by including tolerable pattern variations in size and shape within training data, we expect our autoencoder to judge these variations as tolerable.

Practically, the sub-image size and network are properly chosen so that the selectivity between normal and abnormal patterns is obtained with a sufficient SNR. In the following discussion, we use the clipping size around 2F rather than F, though this makes the range of possible pattern variations more complex than in the above simple example in Fig. 2.

2.2 Procedure of Image Decomposition and Autoencoding

Here, we show the practical procedure of our method using a simple example. In the first step, we generate an autoencoder. SEM images (2048-nm FOV with 2-nm pixel size) are obtained for defect-free 32-nm pitch one-dimensional (1D) random logic patterns fabricated using EUV

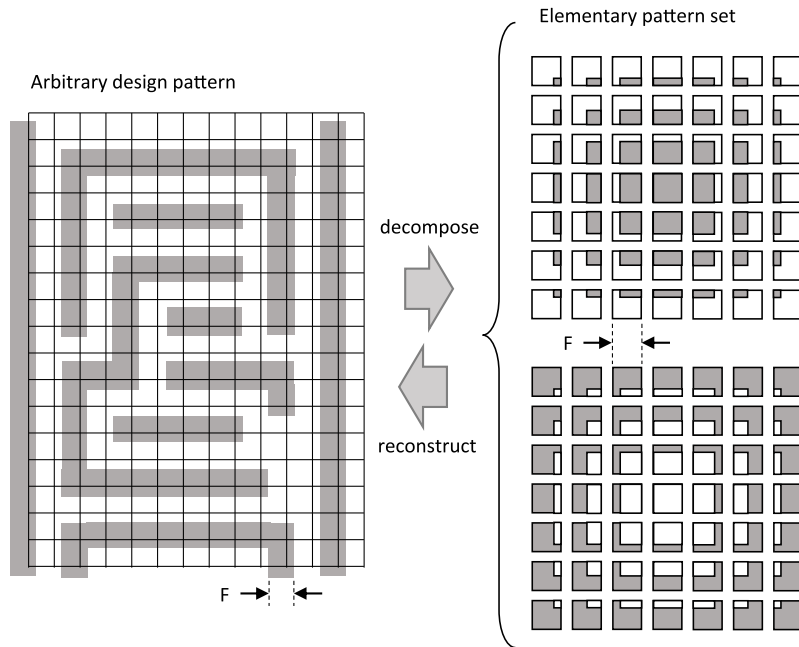


Fig. 2 Example of arbitrary patterns reduced to and reconstructed from elementary pattern set.

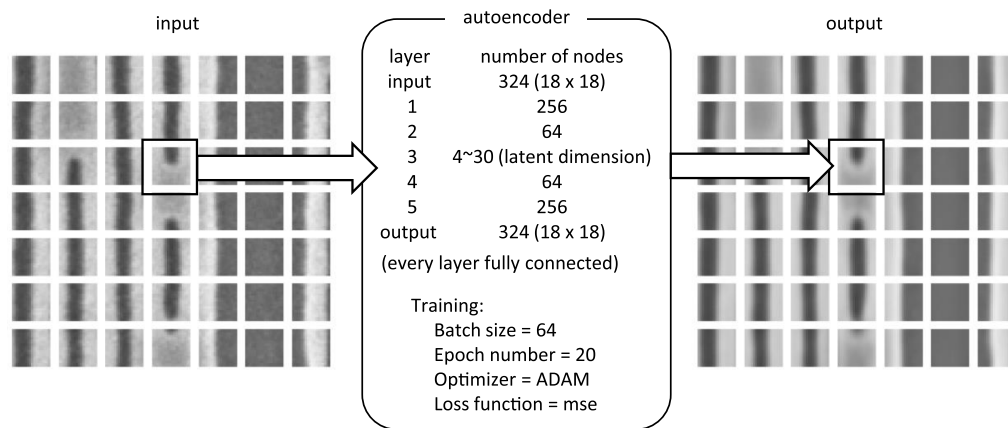


Fig. 3 Autoencoder function for normal patterns. (a) input sub-images, (b) multi-perceptron network configuration, and (c) output of autoencoder after training. The size of sub-images is 36-nm square and the clipping pitch is set equal to the sub-image size for visibility.

lithography (13.5-nm, NA = 0.33). By shifting a 36-nm square clipping window at 10-nm pitch cycle both in the x - and in the y -direction, 677,448 sub-images are extracted from the 18 SEM images, and 75% and 25% of them are used as training and validation data for training an autoencoder shown in Fig. 3. We simply flatten each two-dimensional (2D) sub-image into a 1D vector and apply a simple (seven-layer) multi-perceptron-based autoencoder, which is trained to minimize the loss function defined by the mean square error between input and output images. Details in training conditions are shown in the figure. After training, the autoencoder successfully reconstructs the input sub-images in the output images as shown in Fig. 3. We define a discrepancy index $D(x, y)$ for a particular sub-image at (x, y) by mean square input-output deviation (square sum of deviations between the normalized input image intensity $I_{in}^{(i)}$ for i 'th pixel in a sub-image and corresponding output intensity $I_{out}^{(i)}$ divided by the total pixel number n_{pxl} in a sub-image as shown in Fig. 1). A histogram of the indices for sub-images in training and validation data is plotted by blue in Fig. 4.

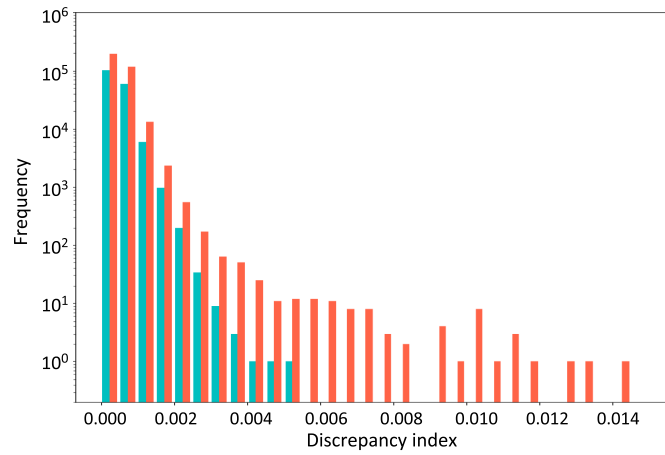


Fig. 4 Histogram of the discrepancy index between input sub-image and autoencoder output for defect-free images (blue) and images including defects (red).

Next, we apply the trained autoencoder to inspect another SEM image taken for the patterns fabricated with the same mask and process as used for the training data, but this time, they contain some defects as shown by a white square in Fig. 5(a) as an example. We inspect 35 image pictures containing various type defects as shown in the left column of Fig. 6.

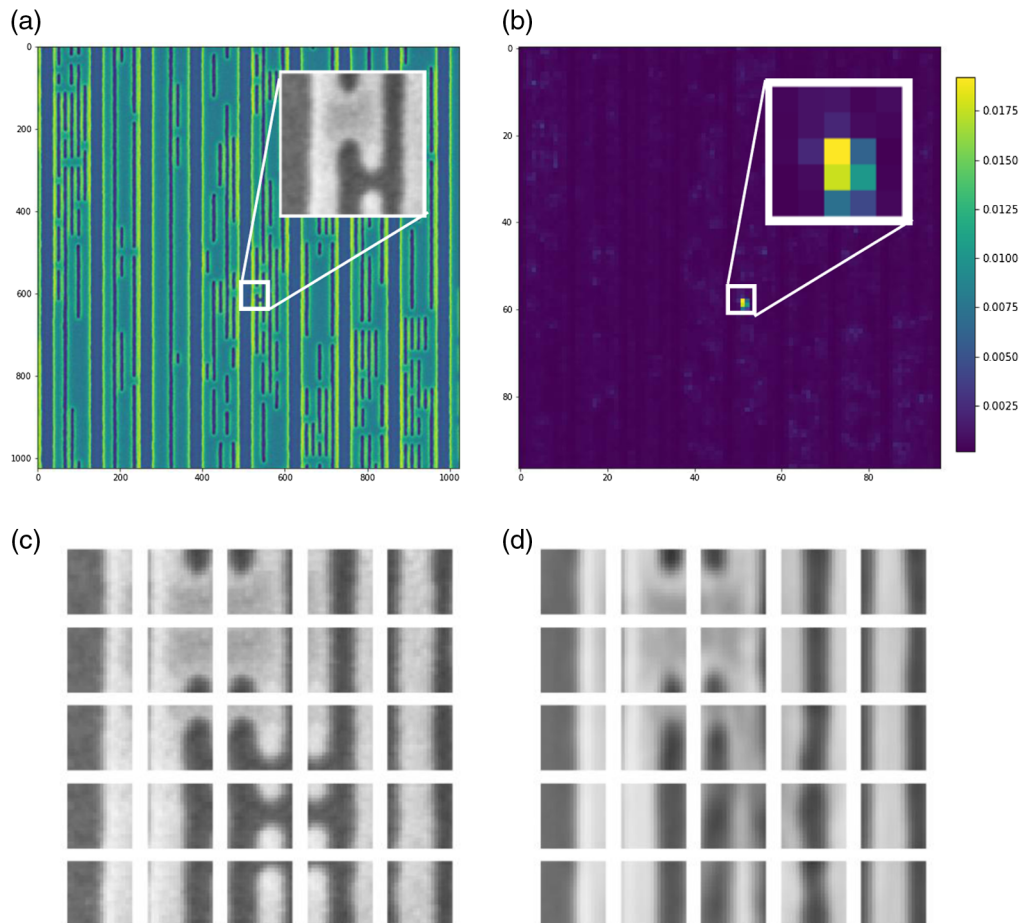


Fig. 5 Autoencoder function for defect patterns. (a) SEM view of typical defect sample. (b) Spatial distribution of the discrepancy index for the sample of (a). (c) and (d) Autoencoder input and output for sub-images around defect. The size of sub-image is 36-nm square and the clipping pitch is 20 nm.

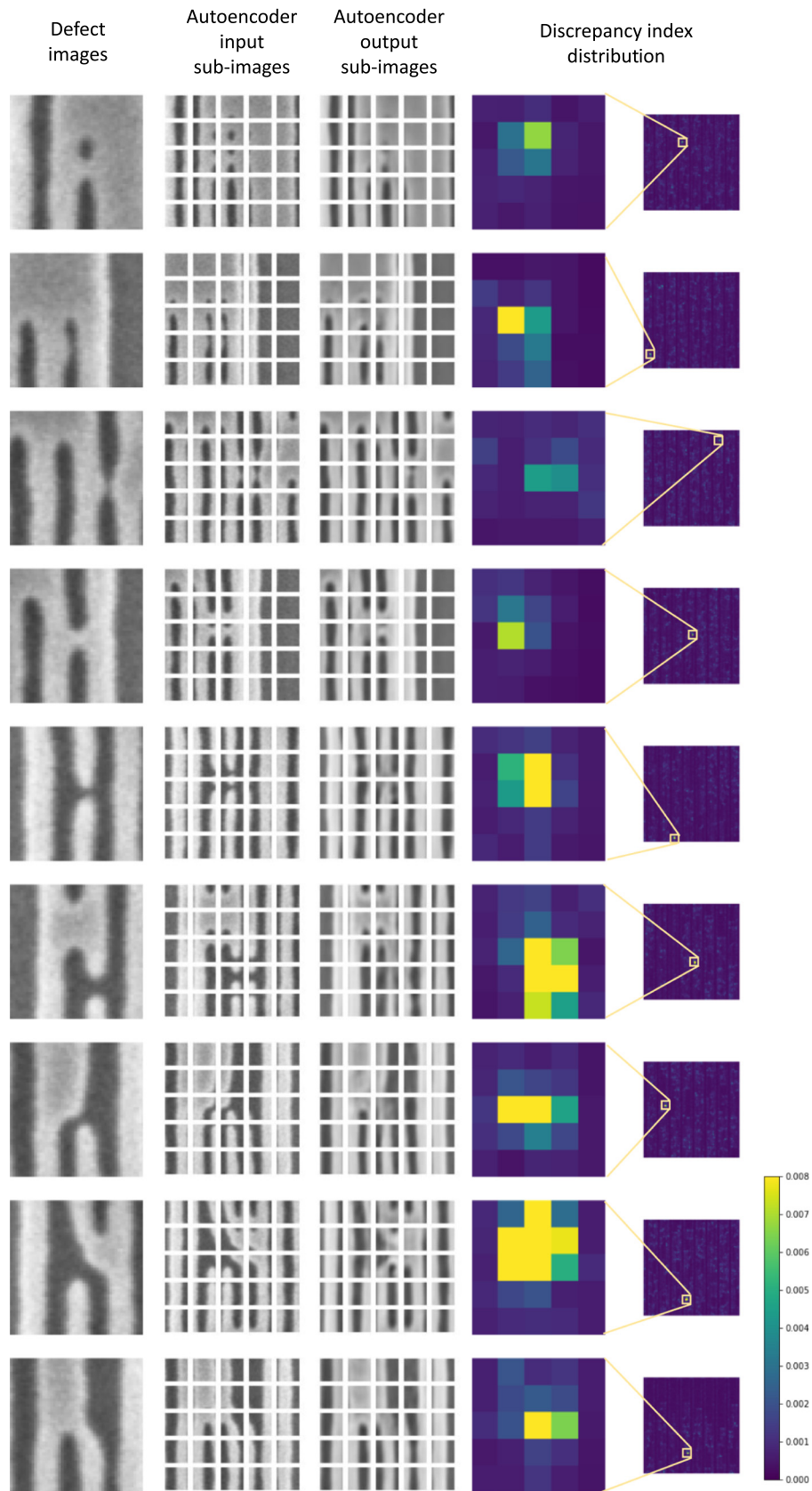


Fig. 6 Autoencoder responses to various type defects. (a) defect examples, (b), (c), and (d) autoencoder input sub-images, outputs, and discrepancy index maps around defects. The size of sub-image is 36-nm square and the clipping pitch is 20 nm.

From these images, the same size sub-images as the training data are clipped in a 20-nm pitch cycle and each sub-image is input to the trained autoencoder. The discrepancy indices $D(x, y)$ are calculated for each sub-image at (x, y) in the inspected images, and its histogram is shown in red in Fig. 4. Although the major parts of the frequency distribution are similar between the defect-free training data (blue) and the inspected data (red), frequency counts appear in a high index range for the tail of the inspected data. In Fig. 5(b), the spatial distribution of the discrepancy index is shown with respect to the sub-image clipping position for the SEM images in Fig. 5(a). We find a high index point at the defect point indicated in Fig. 5(a), and this corresponds to some of high index counts in the histogram. Sub-image inputs and outputs of the trained autoencoder around the above defect are compared in Figs. 5(c) and 5(d). Defects in input sub-images are not reconstructed in the output sub-images while other parts of patterns are reconstructed. We examined the discrepancy between input and output of the autoencoder for sub-images around various type defects shown in the left column of Fig. 6. Although the value of the discrepancy index depends on the relation between the clipping window position and defect patterns, we observe index peaks for every defect as shown in the right column of Fig. 6, and all the defects in the figure are detected by setting the discrepancy index threshold at around 0.003, for example.

Abnormal sub-images are distinguished from normal ones by setting a threshold in the discrepancy index between its distribution ranges for normal and for abnormal sub-images in the histogram. The separation between the above two ranges represents a selectivity of anomaly. We will discuss several factors/parameters for maximizing this separation in the next section.

Time required for autoencoding a full SEM image (1024×1024 pixels) ranges from 0.1 to 0.2 s using a standard laptop PC (i-core 5 with 8 GB memory). This is comparable to or shorter than image acquisition time in most single-beam SEM tools, and thus, it does not limit the throughput of whole inspection processes. Typical training time is 5 to 30 min and both training and autoencoding time are expected to be further shortened by introducing some acceleration devices such as GPUs. The time required for autoencoding a full SEM image is obtained by multiplying the time required for autoencoding a single sub-image and the total number of sub-images for covering the inspection area. The computation time does not depend so much on sub-image size because, with increasing sub-image size, the sub-image autoencoding time increases while the sub-image number for covering the inspection area decreases. It is worth noting that we obtained reasonable results by simply applying multi-perceptron autoencoders to 1D vectors obtained by flattening 2D sub-images within reasonable computation time. In image analysis using deep learning, a convolutional neural network (CNN) is a standard approach today because it achieves record-high performances while keeping the number of network parameters compact. We applied convolutional autoencoders to the same problem as discussed above and obtained results comparable to those with multi-perceptron autoencoders. It took a long time to train the network and to reconstruct sub-images, however, and we find little advantage in using convolutional autoencoders for our purpose here at present.

3 Analysis and Optimization of Autoencoder

3.1 Defect Selectivity and Inspection Pattern Extendibility

A certain sub-image area is necessary for capturing the characteristics of normal patterns to distinguish normal and abnormal patterns, and a certain latent dimension is necessary for encoding normal pattern variations within the sub-image area. The minimum necessary latent dimensions depend on the sub-image size since the number of circuit pattern variations increases with the sub-image size. The optimal sub-image size and latent dimensions depend also on the complexity of patterns to inspect. Here we focus on these two parameters and examine their influences on the autoencoder selectivity to anomalies/defects. We start from simple features toward more complex patterns and compare discrepancy index histograms for normal and abnormal images for several conditions. Conditions other than the above two parameters are fixed for highlighting their influences.

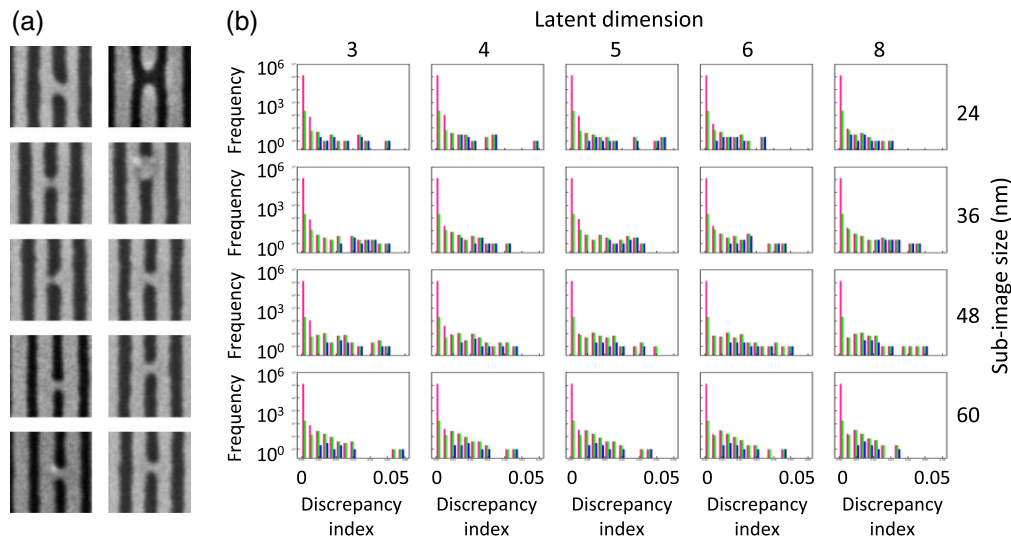


Fig. 7 Discrepancy index histograms of L/S patterns for several sub-image sizes and latent dimensions. Bars in red are for whole image data set including defects, green and blue are for sub-images of vicinity and center of defects shown in (a).

3.1.1 Lines and spaces pattern

First, for simple lines and spaces (L/S), the autoencoder was trained with 10 images each of which contains a defect shown in Fig 7(a) because a sufficient number of defect-free images were not available. The histograms of the discrepancy index, a square sum of the discrepancy between input and output of autoencoder for all pixels in a sub-image, are shown in Fig. 7(b) for autoencoder trained under each different latent dimension and a sub-image size. The histograms for the whole image data set [red in Fig. 7(b)] show clear distribution tails similarly to Fig. 4. Since the area for a particular defect generally spans plural sub-images as shown in Fig. 6, their discrepancy indices span a certain range. Thus, for each defect in each image, 5×5 sub-images including the defect are manually extracted as defect vicinity, and one with the highest discrepancy index of 5×5 sub-images is selected as defect center. Discrepancy index counts of the defect vicinity and defect center sub-images for all images are added to the histograms in Fig. 7(b) by green and blue. With latent dimensions of 5 and a sub-image size of 36 nm, for example, the discrepancy indices for defect center sub-images (blue bars) are higher than 0.01 and they can be detected by setting a threshold around this level. Above this threshold, frequency counts for defect vicinity sub-images (green bars) coincide with that for the whole sub-images in the 10 images (red bars), which means that all the frequency counts above the threshold are from the vicinity of defects. The above means that both the false-negative and the false-positive rates are zero by setting a proper threshold for this particular case in L/S, where all the defects are quite different and distinguishable from normal patterns. In L/S patterns, defect selectivity is obtained with a small sub-image size (24 nm) and low latent dimensions (3), and its dependence on these two parameters is small, though as discussed later, the repeatability in training processes becomes poor for low latent dimensions.

3.1.2 2D array of holes

Results for a 2D array of holes (42-nm pitch) are shown in Fig. 8. Here, again, we trained autoencoders using image data containing defects. From those images, we manually selected several missing holes shown in Fig. 8(a), and for each of them, we extracted the defect center and defect vicinity sub-images similarly to the case in L/S. For small size sub-images (24 nm), discrepancy indices for defect sub-images are buried in main peaks of histograms because for too narrow sub-image, it is difficult to distinguish between normal and abnormal features. For medium size sub-images, the discrepancy indices under low latent dimensions are large both for normal and abnormal images. With increasing the latent dimensions, the discrepancy indices for normal

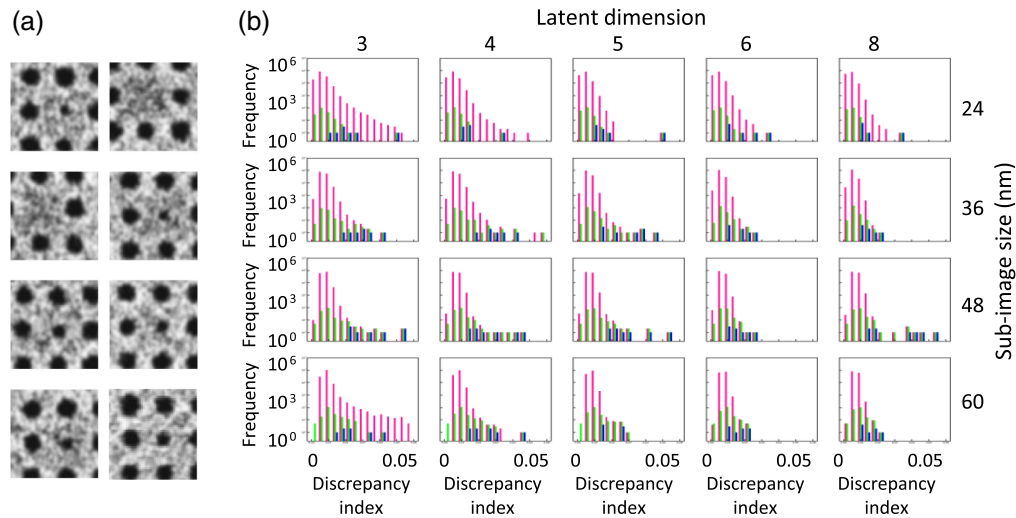


Fig. 8 Discrepancy index histograms of 2D hole array patterns for several sub-image sizes and latent dimensions. Bars in red are for whole image data set including defects, green and blue are for sub-images of vicinity and center of defects shown in (a).

pattern decrease, whereas that for abnormal patterns remains high, enabling us to distinguish between the two. With latent dimensions of 5 and a sub-image size of 48 nm, all the defect centers [blue bars in Fig. 8(b)] can be detected by setting a threshold of 0.015. Immediately above this threshold, the frequency counts for the whole image data set (red bars) are slightly larger than that for defect vicinity (green bars), which means that anomalies other than missing holes are detected. Although they are false positives in terms of missing hole detection, we confirm that they include extremely irregular holes and imbalanced hole sizes among adjacent four holes, which may be worth being detected. Defects in hole arrays are defined as the case where CD is smaller or larger than a certain threshold and consists of a part of CD distribution, and it is difficult to distinguish defect-free distribution and defect distribution, in particular for the samples with large LCDU. Further increasing the latent dimensions decreases the indices for both images, suggesting that the network begins to learn to reconstruct finer structures including defects. In contrast to the case in L/S, the sub-image size and latent dimensions need to be set within a proper range so that the network reconstructs the normal patterns but does not defective images.

To examine the extendibility of the method, we combined the image data for vertical L/S, horizontal L/S (obtained by rotating vertical L/S by 90 deg), and 2D hole arrays, and trained autoencoder with the combined data. The selectivity performance of the trained autoencoder is shown in Fig. 9. Pattern anomalies in each of the above three type features are successfully detected by this single autoencoder by setting the threshold at 0.018 for latent dimensions of 8 and a sub-image size of 48 nm. This suggests that we can expand the variety range of inspected patterns by adding new types of training data with increasing the latent dimensions.

For the above hole-array images (4096-nm square FOVs at 2-nm pixel size), we applied a saturation filter to flatten the intensity higher than 0.75 and renormalized the images. Though this is to remove severe noises observed on unetched surfaces of the original images, which is not related to pattern defects/anomalies of interest, the influence of such pre-cleaning of image data on results needs to be examined carefully.

3.1.3 1D random logic pattern

For 1D-logic patterns, we trained autoencoders using defect-free image data and examined their responses to defect images with the same data as used in Sec. 2.2 under several different latent dimensions and sub-image sizes. We manually selected 35 typical defect samples in 35 image data (shown as “labeled defects” in Fig. 11) and extracted the defect center and defect vicinity sub-images for each of them. Histograms for the whole image data set containing defects (red in

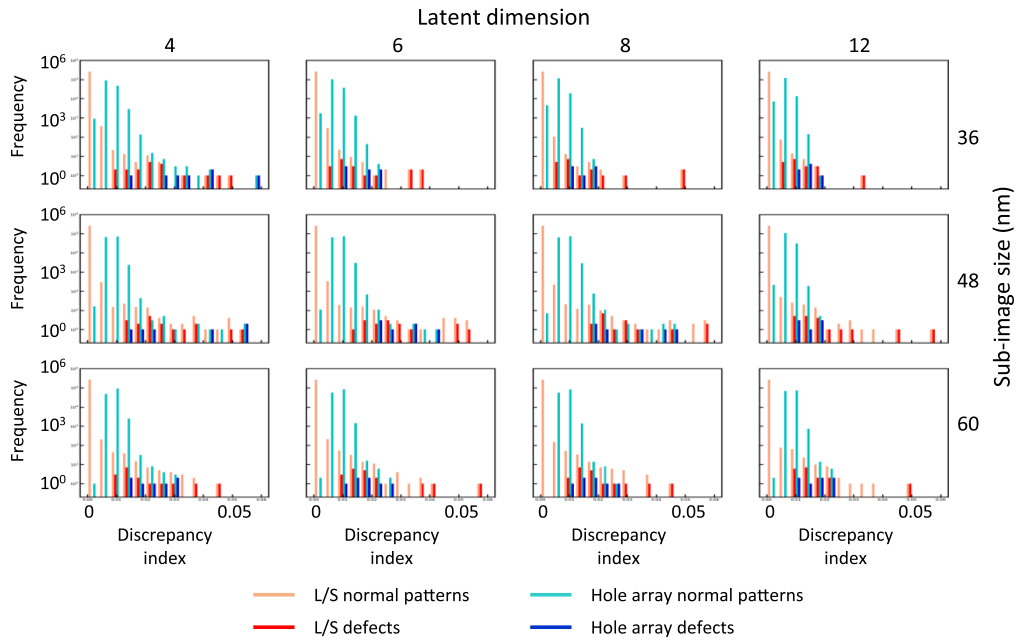


Fig. 9 Discrepancy index of autoencoder trained by mixed-type patterns. Histograms are shown separately for each-type pattern and defect.

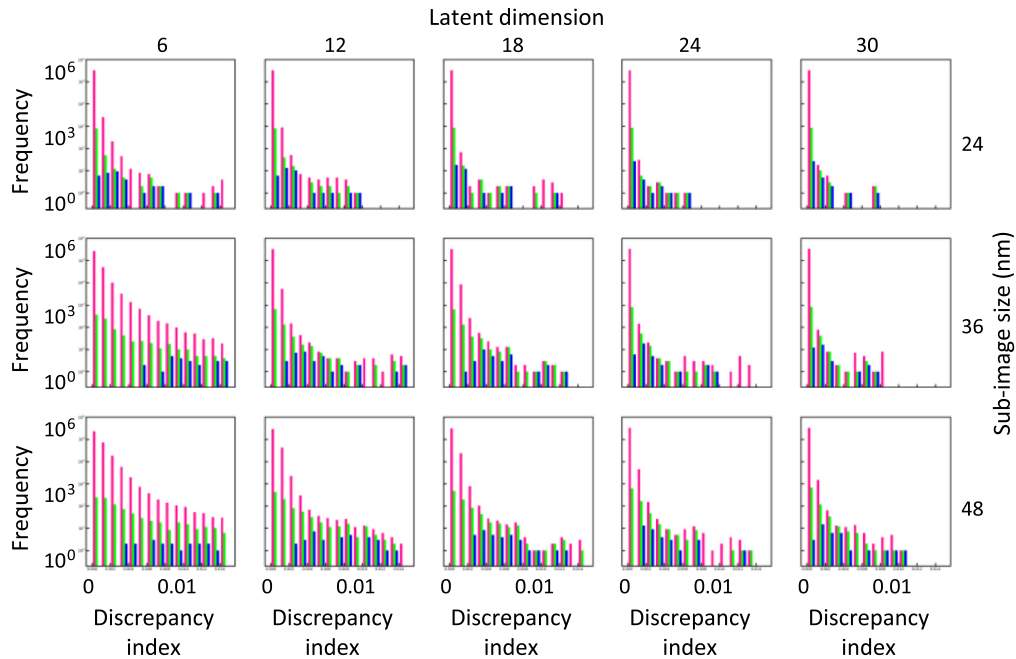


Fig. 10 Discrepancy index histograms of 1D random logic patterns for several sub-image sizes and latent dimensions. Bars in red are for whole image data set including defects, green and blue are for defect vicinity and center sub-images.

Fig. 10) show clear distribution tails similar to the case in a 2D array of holes (Fig. 8). For small sub-image size (24 nm), discrepancy indices for defects are buried in the main peaks of histograms because too narrow sub-image is difficult to distinguish normal and abnormal features. Discrepancy index rapidly increases with sub-image size since the pattern variations within the sub-image increase with sub-image size, and higher latent dimensions are required for encoding them. For 36-nm sub-images, the discrepancy indices under low latent dimensions are large both for the normal and for the abnormal images because networks cannot well reconstruct even

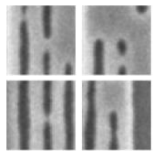
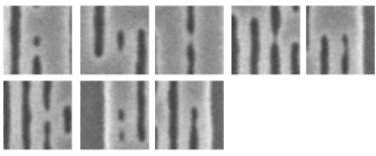
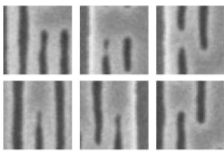

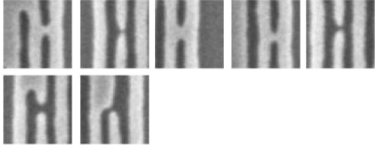
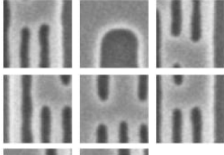
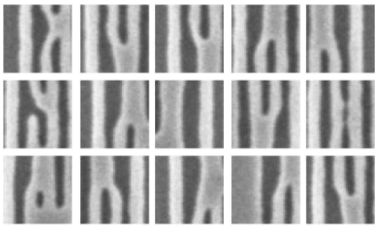
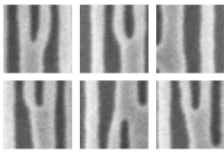
	Labeled (35 typical anomalies sampled)		Unlabeled
	Undetected	Detected	
Quasi-missing- Necking- Narrowing- trench			
Breaking- Broadening- line			
Bridging- line			

Fig. 11 Classification of detected and missed pattern anomalies. Note that labeled anomalies are typical defect samples arbitrarily selected from the inspected data and are not the complete list of true defects.

normal images and it is difficult to distinguish them. The discrepancy indices for normal images decrease with increasing the latent dimensions, showing that the network learns to reconstruct the normal pattern images. Since it still cannot reconstruct abnormal images, however, the discrepancy indices remain high for defective images, enabling us to distinguish the two. Further increasing the latent dimensions decreases the indices for both images, suggesting that the network begins to learn to reconstruct finer structure including defects. With latent dimensions of 18 and a sub-image size of 36 or 48 nm, most defect centers for the 35 labeled defects (blue bars in Fig. 10) are detected by setting a threshold of 0.003. Above this threshold, the frequency counts for the whole image data set (red bars) slightly exceed that for defect vicinity (green bars), suggesting that the image data contain anomalies other than the 35 labeled defect samples. To show the nature of detected anomalies more specifically, Fig. 11 categorizes detected or labeled defects into three groups, detected labeled, undetected labeled, and detected unlabeled anomalies for three defect-types. Note that the above three groups don't represent true-positives, false-negatives, and false-positives, respectively, but provide a snapshot of detection capability under a specific detection threshold (0.003) since the 35 labeled defects do not represent all the true defects contained in the evaluated 35 images. For bridging, broadening, and breaking types, most labeled defects were detected, and several other defects that are similar but not labeled were newly found. In contrast for quasi-missing, necking, and narrowing types, some labeled defects were not detected while several other defects that are similar but not labeled were found. These undetected labeled anomalies are detected by lowering the threshold with the increased number of detections in quasi-missing, necking, and narrowing type (unlabeled) defects, suggesting the above threshold is appropriate for bridging, broadening, and breaking-type defects but too high for quasi-missing, necking, and narrowing-type defects. Note that completely missing pattern features are difficult to detect without reference design data. We will discuss varieties of pattern anomalies detected in 1D-logic patterns in Sec. 4. The sub-image size and the latent dimension need to be set so that the former covers the necessary pattern variations while keeping enough defect selectivity, and the latter reconstructs the normal patterns but does not defective images.

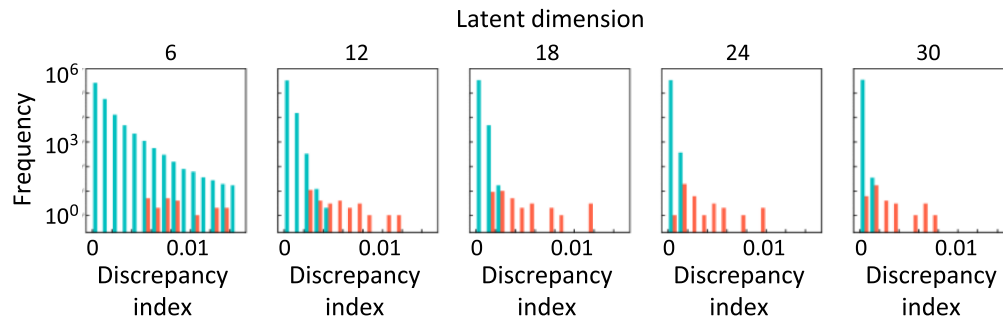


Fig. 12 Discrepancy index histograms of quasi-2D random logic patterns for several latent dimensions. Blue and red are for normal pattern images and for defect center sub-images.

With further examination of the extendibility of the method, we mimic more complex 2D logic IC patterns by adding rotated images of 1D-logic since 2D logic images are not available. The images used for evaluating the 1D-logic autoencoders are rotated by 90 deg and are added to the original data for training quasi-2D logic autoencoders. Trained autoencoder successfully reconstructs normal data in both directions and distinguishes defects from normal patterns by setting the latent dimensions around 24 (Fig. 12). This again shows that we can expand the variety range of inspected patterns by adding new types of training data with increasing the latent dimensions, although the training data for real arbitrary 2D design circuits need other features such as corners and crossings in resist lines and trenches.

3.1.4 Tuning and validation of autoencoder configuration

Although one of the advantages of autoencoders is that it requires only normal data for training as previously mentioned, in this section we used abnormal data for tuning and validating the autoencoder configuration. In general, the size of anomaly data for this tuning/validation purpose is much smaller than that required for training DNN with anomalies. Since defect-free data and defect data as used for explaining the method in Sec. 2.1 are not always available, in practice, we train and tune autoencoder with data that may contain some defects. Since the occurrence of defects is rare, we expect that autoencoder can be trained mainly for a normal and majority part of data. We search autoencoder configurations showing tails in discrepancy index histogram and further tune the configuration so that defects of interest are included within the tail. Although we expect that the tuned autoencoder detects other types of anomalies than that used in the tuning process, this is not guaranteed. To approach this problem theoretically, we will attempt to use variational autoencoder (VAE) at the end of this section (Sec. 3.3). As a more practical approach, however, we first tuned/validated autoencoder with small anomaly data and apply it to larger data containing broader type anomalies and examine its effectiveness. In Sec. 4, for example, we will apply an autoencoder whose configuration we tuned above using smaller data samples (35 images including typical defects as shown in Fig. 11) to much larger size data. For further quantitative evaluations with false negative/positive rates, cross-validation with some reference inspection methods is required with a clear definition of defect criteria, and this is left for future work.

3.2 Repeatability and Over-Fitting in Network Training

Convergence and repeatability in training autoencoder are examined. Autoencoders with several latent dimensions were trained repeatedly (five times) using the same training data, and for each trial, loss-trend curves (for training and validation data) and discrepancy index histograms after 20 epochs are shown in Figs. 13(a) and 13(b). They show a large variation when the latent dimensions are low probably due to random initialization of weights and data split. Thus, to obtain fair results within practical epoch number, it is generally desired to try training several times to confirm the repeatability of results or for choosing the best-performing network. The results in Figs. 7–12 were selected from the most favorable results in several trials for each condition.

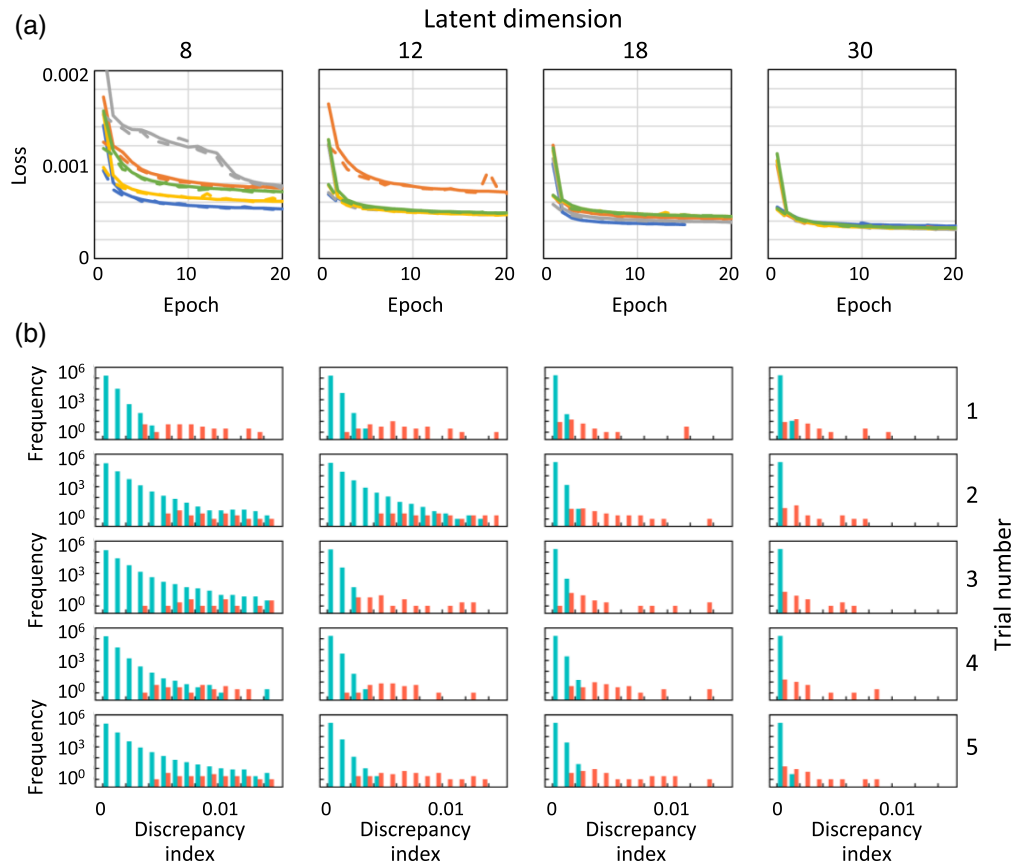


Fig. 13 Dependence of convergence and repeatability in training process on latent dimensions. Training of 1D-logic pattern images is repeated five times. (a) Relationship between loss function and the number of epoch in the training process. Solid and dashed lines are for training and validation data, and different colors are for different trials. (b) Discrepancy index histograms of 1D random logic patterns. Blue and red are for normal pattern images and defect center sub-images.

In the above examples, the typical number of sub-images used for training data is 1000 k, which was limited by the computer resource used in this study. To avoid over-fitting, this number needs to be sufficiently large compared with the number of network parameters (100 k to 400 k typically), and the above training data number may not seem enough. However, the decreasing trend in loss functions for training data and validation data shows no sign of overfitting as shown in Fig. 13(a). It has been known that injection of noise suppresses over-fitting and improves the performance of supervised and unsupervised neural networks including autoencoders.¹¹ Our input data, SEM images are known to contain a considerable amount of noise inherent in its image formation mechanisms, such as stochastic electron emission and scattering within a specimen, and this may effectively suppress over-fitting. Also, we found no particular effectiveness in introducing regularization or dropout.

3.3 Fundamental Reliability

Although the above discussion shows the practical effectiveness of the method, a fundamental question remains: is it guaranteed that autoencoder does not reconstruct defect images? One approach to answering this question is to show that defect images are not contained in the space generated by the decoder. Since it is difficult for multi-perceptron autoencoders to visualize this, however, here we apply VAE. In VAE, an encoder encodes an input image to a vector (z_1, z_2, \dots, z_n) in an n -dimension latent space, and a decoder generates an output image from a nearby vector $(z_1', z_2', \dots, z_n')$ in the latent space. This is similar to conventional autoencoders, but in VAE, similar input patterns are encoded to points close to each other in the latent space,

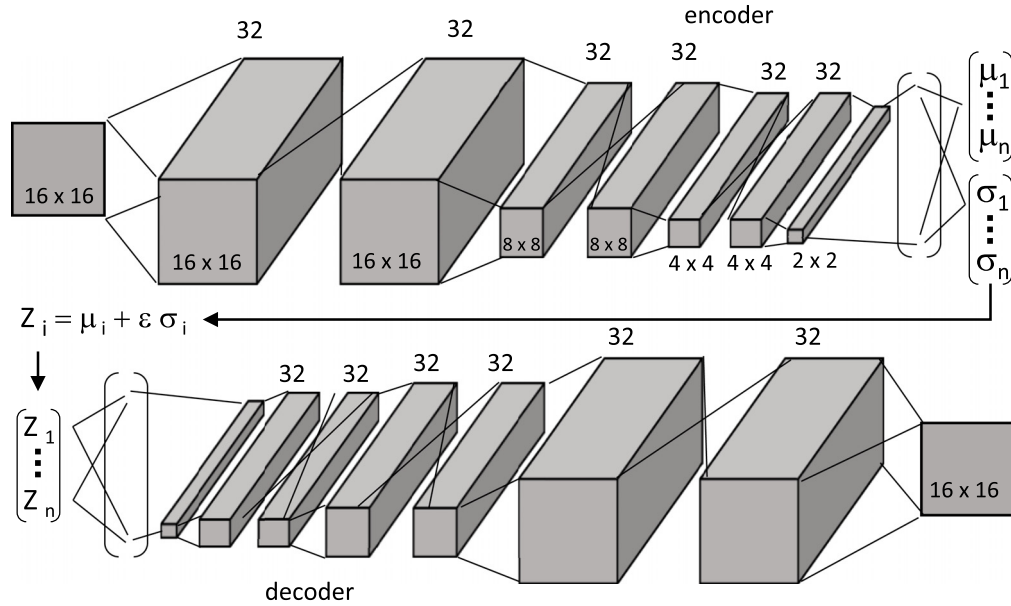


Fig. 14 Convolutional variational autoencoder (CVAE) configuration.

and from nearby points in the latent space, the decoder generates output patterns similar to each other. See Refs. 12 and 13 for more details on VAE. Any input image is encoded to some point in the latent space no matter if it contains defects or not. Thus, by examining variations in images generated from every point of latent space, we can examine possible variations in the output of the decoder.

Convolutional VAE ($n = 2$) shown in Fig. 14 is trained with the 2D hole array data used previously in Sec. 2.1. Figure 15(a) shows the decoder output images generated from each point (latent vector) of 2D latent space (z_1, z_2) after training. Generated images cover all the possible variations in patterns within the window size of the sub-image but contain no defect image. Here, we used a simple 2D hole array since it requires low dimension (2) for latent space for covering the pattern variation, and its effectiveness is easily visualized as shown in Fig. 15.

Next, we input sub-images around defects shown in Fig. 15(b) into the trained VAE. Sub-image clipping pitch (20 nm) was set slightly smaller than half pattern pitch (21 nm). The latent vector encoded from each sub-image ($m_i = \text{mean component of } z_i$) is plotted in Fig. 15(c), and the sub-images generated by VAE are shown in Fig. 15(d). We categorize 36 sub-images with nine holes into four types A, B, C, and D as shown in Fig. 15(b), and latent vectors for the same type form a cluster on the latent space, including defect hole shown by red in Figs. 15(b) and 15(c). Similar sub-images are generated from the latent vectors for each cluster [Fig. 15(d)], and again, we can detect a defect hole by comparing the input and output sub-images. Note that the generated images reproduce small differences in hole pattern positions due to the difference between clipping and pattern pitches.

Using properly trained VAE, it is unlikely that autoencoder reconstructs defect images although this is not the verification for multi-perceptron autoencoder. Unfortunately, VAE requires longer computation time than multi-perceptron (both for training and for reconstruction), and detection accuracy of multi-perceptron needs to be examined from the requirement for IC pattern inspection, as reliability is often the issue in applying AI to mission-critical problems.

4 Application Results

4.1 Experiments

Here we examine the feasibility of our method in detecting anomalies in moderate areas of IC patterns fabricated by EUV lithography. First, 30- μm square areas of 1D design random logic

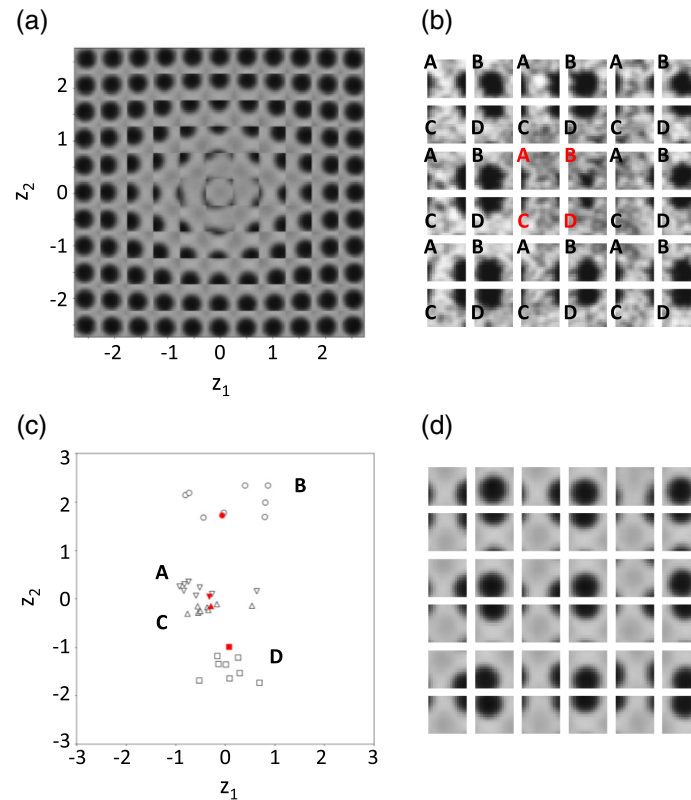


Fig. 15 Convolutional variational autoencoder. (a) Decoder output images generated from each point of 2-dimensional latent space (z_1, z_2); (b) about 36 sub-images around a missing hole defect; (c) distribution of latent vectors encoded from 36 sub-images in (b); shown in red are for sub-images including the defect; (d) CVAE output sub-images corresponding to (b).

circuit patterns with 32- and 36-nm pitch were inspected. A mask containing the block of the above two kind patterns was exposed by EUV exposure tool ($NA = 0.33$) at eleven exposure fields, F1, F2, . . . , F11. Each $30\text{-}\mu\text{m}$ square area of the 22 blocks (32- and 36-nm design-pitch blocks in fields F1 to F11) was inspected by an SEM inspection tool,⁵ and 225 pictures (2048-nm square FOV with 2-nm pixel size) were obtained for each block.

In the previous examples, we used normal data for training autoencoders. Since another analysis using the same image data⁵ shows that the best patterning results are obtained for the field F5 to F9, we trained autoencoders using the image data for these fields. For each design pitch, we clipped 36-nm square sub-images from each of 675 images (F5, F7, and F9) with 10-nm clipping pitch in both x - and y - directions, and from about 12,500,000 clipped sub-images, about 1,000,000 sub-images were selected as training data. This number of training data is limited by computing resources, and a larger number is desirable in general. In real IC manufacturing environments, defect-free images are not guaranteed for training data, and we do not confirm our training data to be defect-free. Since we generate the training data by selecting a part of data within the process window where the defect probability is generally low, we expect the number of defects contained in selected training data is small. Further, even if they contain some defects, as the general characteristics of autoencoders, they rarely learn to reconstruct defect images in general if the number of defects in training data is sufficiently small compared with that of normal patterns as we saw in Figs. 7, 8, and 9.

Next, we applied the above-trained autoencoders to all the SEM pictures taken for 11 fields for both design-pitches and calculated the discrepancy indices for every sub-image within $30\text{-}\mu\text{m}$ square pattern blocks for each field. If an end of normal pattern in an adjacent area appears in the periphery of a sub-image, autoencoders sometimes detect it as an anomaly. This can be judged by shifting the sub-image window position to contain the whole part of interest. Thus, to avoid this

false detection, the clipping windows are set so that neighboring sub-images have overlapping areas, and we count the anomalies detected in adjacent sub-images as one identical defect. As a result, detected anomalies are classified using two parameters, discrepancy index representing the strength of anomaly and multiplicity (the number of sub-images related to one defect) representing the size of anomaly and/or the reliability of detection.

4.2 Defect Characteristics

Spatial distributions and frequency distributions (histograms) of discrepancy index for sub-images contained in 30- μm square block are shown in the second and fourth columns of Figs. 16 for 11 exposure fields. Frequency counts for high discrepancy index increase as the exposure fields deviate from their center. We set a threshold for the discrepancy index and extracted high index spots as anomalies/defects. The distributions of extracted defect locations are shown in the third column of the figure. Detected anomalies are classified using the previously explained discrepancy and multiplicity indices. Scatter plots between these two indices are also shown in the fifth column of Fig. 16.

For lower field numbers, the number of defects rapidly increases with decreasing the field number, and their multiplicity and discrepancy distribute are limited. The SEM images around the extracted defect spots are shown in Fig. 17. We find missing/necking-type defects for 32-nm design-pitch and many edge-placement-error (EPE) type defects such as local narrowing of trench width for both design-pitches. We also detect some distortions or deformations in local pattern shapes that are difficult to classify. For higher field numbers, the number of defects rapidly increases with the exposure number, and a positive correlation is found between discrepancy and multiplicity (Fig. 16). For anomalies with high discrepancy and multiplicity indices, we find bridging-type defects in particular for 32-nm design pitch, which is supposed to be from the local collapse of narrowed resist lines (Fig. 17). For 32-nm pitch samples, we detected a few defects for both types even at the center field (F7). Anomalies with low discrepancy and multiplicity are mostly EPE-type and more in number than the bridging-type defects. The product of both indices can also be used for screening and classifying anomaly types. Again, we detected a lot of distortions or deformations in local pattern shapes that are difficult to classify. Although they are not critical defects such as necking or bridging, their impact should not be overlooked, and the present method effectively detects these unknown anomalies with neither prior knowledge about them nor the need of defining them in advance. It clarifies the dependence of the number and type of defects on exposure fields using a single measure.

4.3 Discrepancy Index as Patterning Process Monitor/Pattern Fidelity Indicator

Since the discrepancy index reflects any deviation in pattern image from ideal, expected, or allowed ones, we can use it as a single indicator for varieties of problems in patterning processes and resultant pattern fidelities. Further, their statistical information such as the frequency and spatial distributions help us reliably capture a tiny sign of these problems from complex unknown causes, and we can use them beyond defect inspection purposes.

For example, the shape of the discrepancy index histogram shown in Fig. 16 is very sensitive to the change in processes even below the defect criteria. Thus, it can be used as a predictor/warning of problems in patterning processes even if it is for a limited area and no defect is detected in the area. The spatial distribution signatures of the discrepancy index also effectively visualize defect characteristics and sometimes suggest their origin. The spatial distributions of the discrepancy index within 30- μm square pattern block (Fig. 16) and its typical magnified image within 2- μm square FOV [Fig. 18(a)] show that high discrepancy index spots appear randomly, suggesting their generation mechanism is stochastic. In another sample, however, we observe a clear cluster of anomaly spots, which could suggest large size contaminants or particles [Fig. 18(b)]. Another example shown in Fig. 18(c) observed for the sample after etching clearly visualizes the damage on resist patterns caused by SEM observation after a resist development.

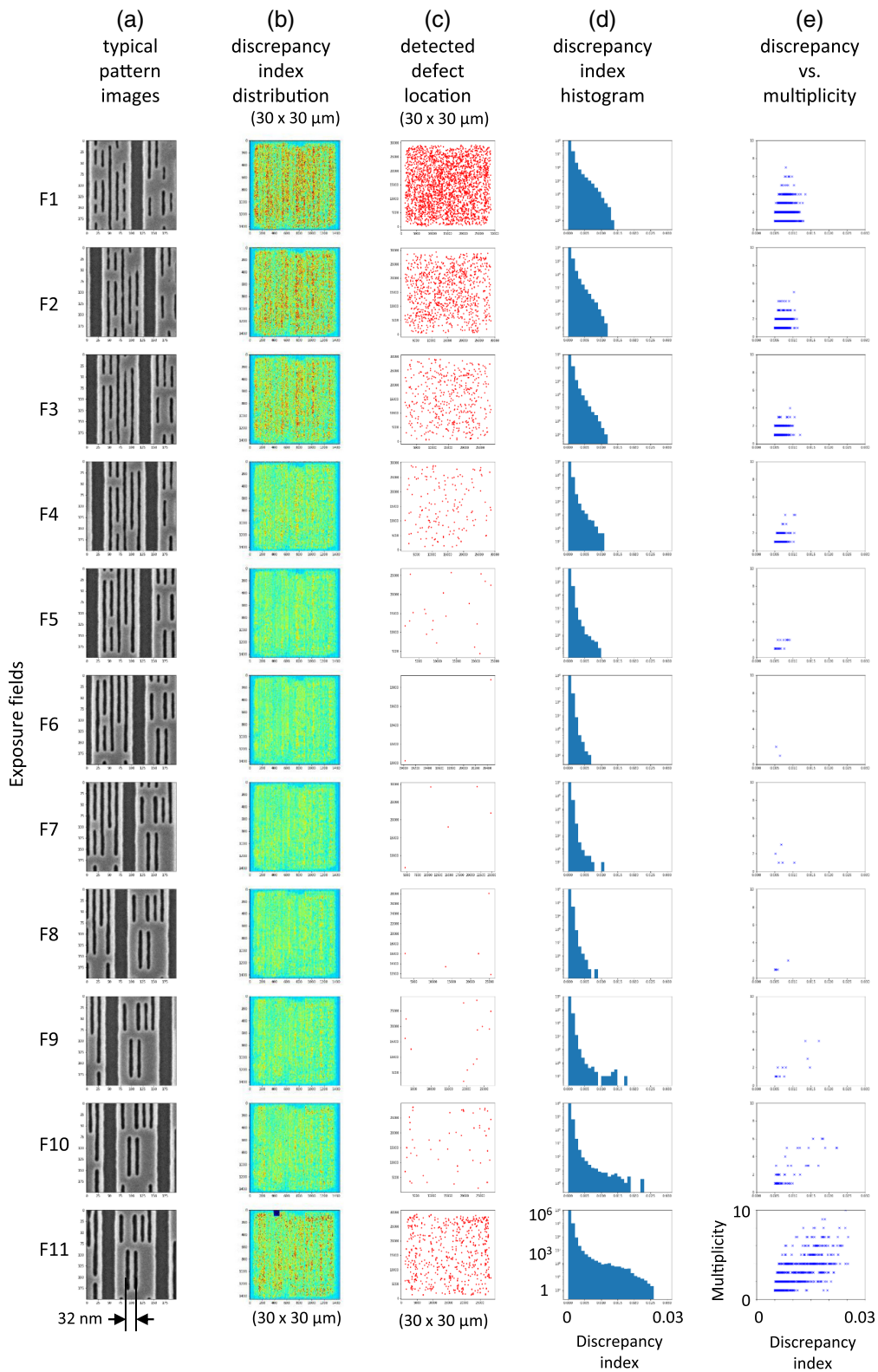


Fig. 16 Defects/anomalies in 1D random logic patterns detected by autoencoder for several exposure fields (F1–F11). (a) SEM views of typical sample patterns; (b) and (c) distributions of discrepancy index and defect locations in 30- μm square inspected area; (d) histograms of discrepancy index for sub-images within 30- μm square inspected areas; and (e) distribution maps of discrepancy and multiplicity indices for defects detected within 30- μm square inspected areas.

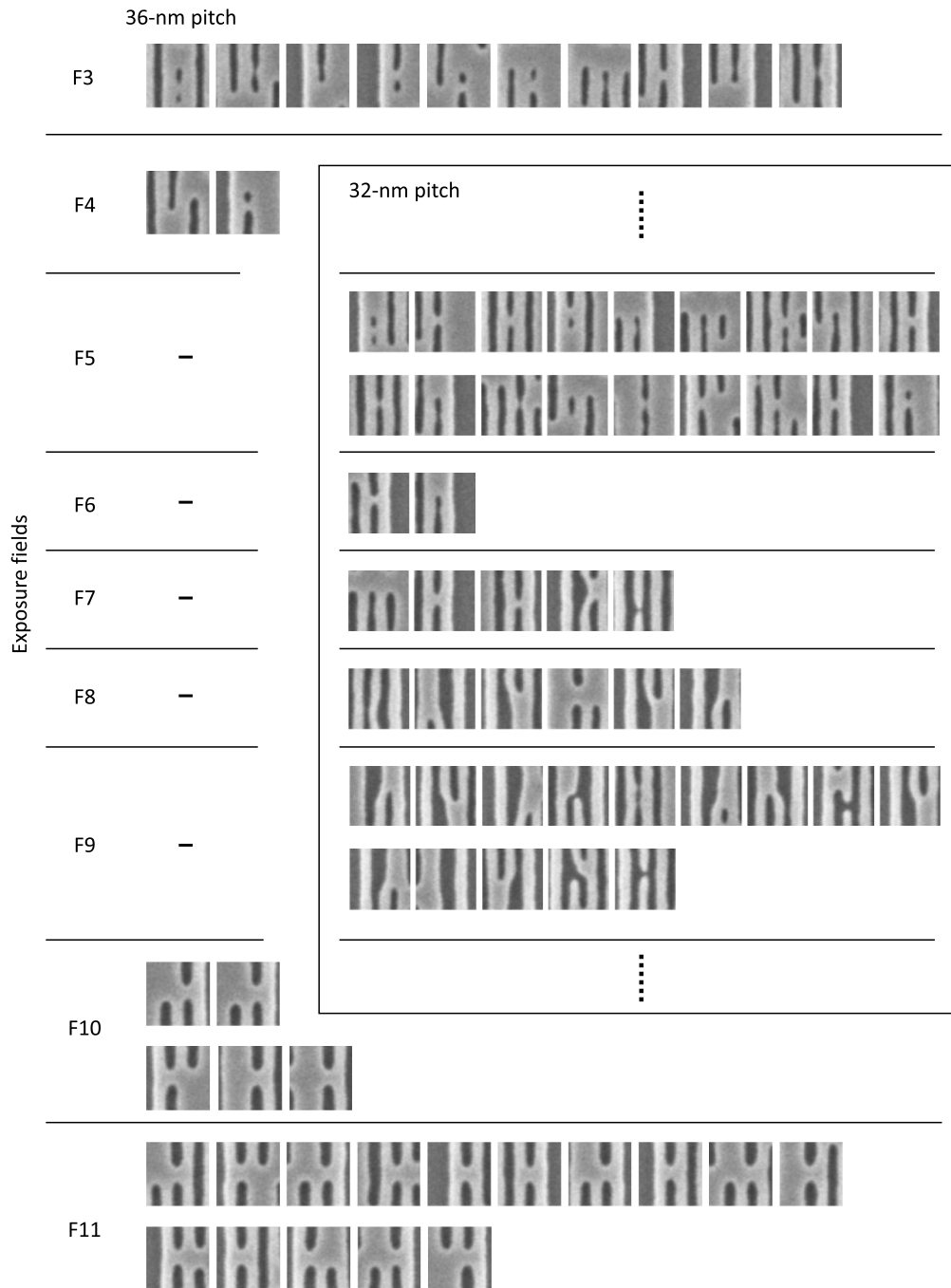


Fig. 17 Magnified SEM images of defects detected by autoencoder.

4.4 Defect Sensitivity

Lastly, we show the ability of the method to detect extremely tiny defects. For defects smaller than sub-image size, the ratio between the sizes of defects and normal patterns increases with decreasing sub-image size, and thus, a high SNR defect detection is expected as we previously explained. Here, 2D arrays of pillar patterns are inspected by the autoencoder trained and optimized for the patterns using image data containing defects. The histograms of the discrepancy index [Fig. 19(a)] show clear distribution tails and we extract sub-images with the discrepancy index higher than 0.003 as defects. Spatial distributions of the discrepancy index [Fig. 19(b)] and magnified SEM images around extracted defects [Fig. 19(c)] are shown for each detected defect. The autoencoder detects an extremely tiny (several nanometers in diameter) defect in

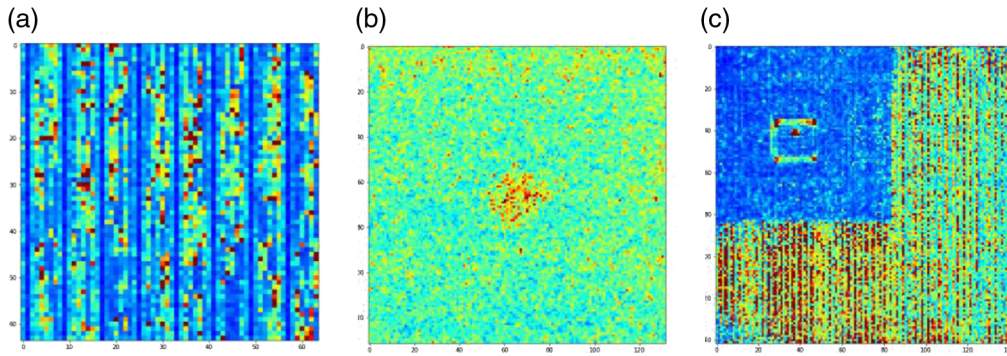


Fig. 18 Spatial (across-FOV) distributions of the discrepancy index. (a) randomly generated defects; (b) clustered defects; (c) SEM damage. The sizes of FOV are 2, 3, and 3 μm for (a), (b), and (c).

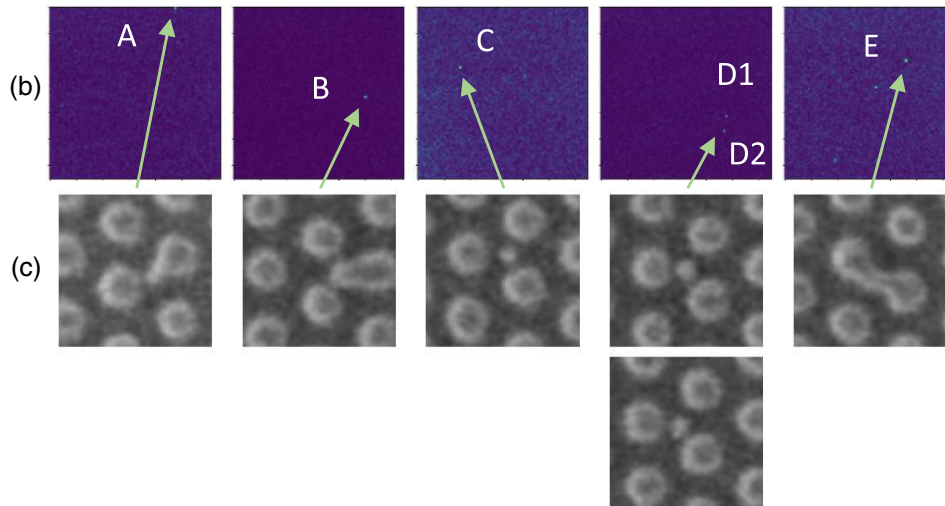
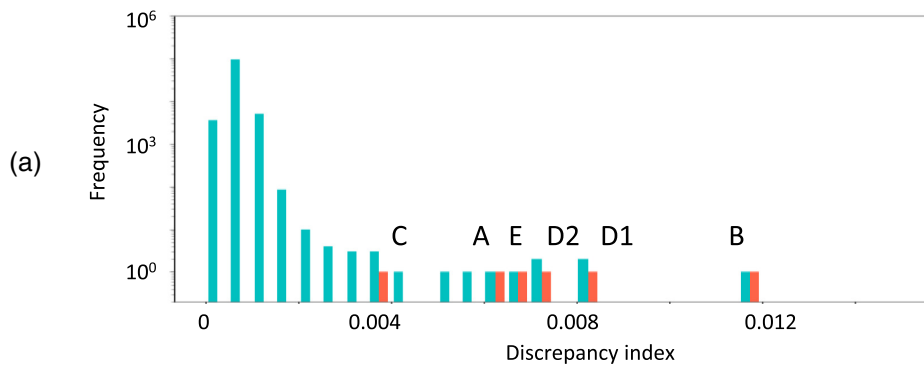


Fig. 19 Defects in a 2D array of dots detected by autoencoder. (a) Discrepancy index histogram; (b) across-FOV distributions of discrepancy index containing defects A–E; and (c) magnified SEM viewgraph around each extracted defect.

Case C and D with high SNR as shown in Fig. 19(b) as well as a typical pattern defect in Case A, B, and E.

5 Conclusions

The method presented in this paper will be effective and useful in particular when a huge amount of image data becomes available from high-speed SEM inspection tools for advanced EUV

patterns and pattern anomaly detection is required for them. It enables us not only to directly detect a wide variety of defects with neither design data nor prior knowledge about defects but also to capture a small sign of change in process conditions and pattern fidelity through frequency and spatial distributions of autoencoder discrepancy index.

Acknowledgments

The authors acknowledge IMEC for the sample preparation and their support of this work.

References

1. S. Hunsche et al., "A new paradigm for in-line detection and control of patterning defects," *Proc. SPIE* **9424**, 94241B (2016).
2. P. De Bisschop, "Stochastic printing failures in extreme ultraviolet lithography," *J. Micro/Nanolithogr. MEMS MOEMS* **17**(4), 041011 (2018).
3. V. Anantha et al., "Optical defect inspection solution for EUV stochastics detection," *Proc. SPIE* **11323**, 113231J (2020).
4. E. Ma et al., "Multi-beam Inspection (MBI) development progress and applications," *Proc. SPIE* **11325**, 113250F (2020).
5. T. Kondo et al., "Massive metrology and inspection solution for EUV by area inspection SEM with machine learning technology," *Proc. SPIE* **11611**, 1161111 (2021).
6. G. Pang et al., "Deep learning for anomaly detection: a review," *ACM Comput. Surv.* **54**, 1–38 (2020).
7. C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proc. 23rd ACM Int. Conf. Knowl. Discovery and Data Mining*, pp. 665–674 (2017).
8. H. Lei et al., "Automated wafer defect classification using a convolutional neural network augmented with distributed computing," in *31st Annu. SEMI Adv. Semicond. Manuf. Conf.* (2020).
9. B. Dey et al., "Unsupervised machine learning based CD-SEM image segregator for OPC and process window estimation," *Proc. SPIE* **11328**, 113281G (2020).
10. A. Baranwal et al., "A deep learning mask analysis toolset using SEM digital twins," *Proc. SPIE* **11518**, 1151814 (2020).
11. B. Poole, J. Sohl-Dickstein, and S. Ganguli, "Analyzing noise in autoencoders and deep networks," arXiv:1406.1831 (2014).
12. D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," arXiv:1312.6114 (2013).
13. A. Jaan, "Tutorial - What is a Variational Autoencoder?," Zenodo (2016).

Hiroshi Fukuda joined Hitachi Central Research Laboratory in 1985, where he has engaged in various fields of lithography as well as nano-devices, MEMS, and HDD, including research activities in Stanford University and Hitachi Europe Ltd. He has been with Hitachi High-Tech since 2012. He received his BS, MS, and PhD degrees from Tokyo Institute of Technology in 1983, 1985, and 1994, respectively. He has published about 40/90 journal/conference papers and holds over 20 patents. He is a fellow of SPIE.

Tsuyoshi Kondo: Biography is not available.