Master of Science in Mechanical Engineering

Effective Data Selection for Robust Training of Generative Adversarial Network for Lithography Pattern Alignment

리소그래피 패턴 정렬용 생성적 적대 신경망의 강건한 훈련을 위한 효과적인 데이터 선택

February 2023

Graduate School of Mechanical Engineering
Seoul National University

Nohong Kwak

Effective Data Selection for Robust Training of Generative Adversarial Network for Lithography Pattern Alignment

Advisor Do-Nyun Kim

Submitting a master's thesis of Mechanical Engineering

October 2022

Graduate School of Mechanical Engineering
Seoul National University

Nohong Kwak

Confirming the master's thesis written by Nohong Kwak

December 2022

Chair	윤 병 동	(Signature)
Vice Chair	김 도 년	(Signature)
Examiner	양 진 규	(Signature)

Abstract

Critical Dimension SEM (CD-SEM) is a dedicated system for measuring the shape, size and roughness of patterns formed on semiconductor wafers. As designs shrink and product development challenges increase, the ability to quickly measure large amounts of samples for accurate Optical Proximity Correction (OPC) is required. Design Based Metrology (DBM) technology allowed the rapid creation of large volumes of recipes using design images and reduced measurement time. However, there were still many problems in the alignment between the design image and the SEM image, and to solve this problem, a new pattern alignment method using Generative Adversarial Network (GAN) technology was developed. In this paper, training patterns are classified according to polygon types of design patterns and the alignment effect according to each type is confirmed. We also studied how to effectively select a training set for model training through the relationship between training set and alignment accuracy.

Keyword : Scanning Electron Microscopy, Design-based metrology, Generative Adversarial Network, Supervised learning

Student Number : 2021-24764

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

1.1. Semiconductor design and optical proximity correction

Semiconductor design begins by with setting the specifications of a chip and explaining the operation of the chip, and appears as a logical design expressing the connection relationship between gates, which are the basic components of a semiconductor. The verified logical design is converted into a physical layout through Place & Route (P&R)[1]. P&R is the process of placing a library and wiring it to connect, which results in very complex geometries as it has to navigate the placement and routing paths that satisfy given design constraints. The P&R process creates a physical drawing, which is saved as a Graphics Design System (GDS)[2] format file. This process is called Physical Synthesis.

GDS files express the physical shape of semiconductors using layers and polygons. A GDS file is a format similar to a vector graphic file, but it is composed of basic geometric elements such as rectangle, polygon, and path, excluding complex elements. A set of these elements is defined in a unit called a cell, and a complex drawing is expressed with a small size of data by repeatedly placing cells or organizing them into hierarchies[3].

It is very difficult to express a complexly designed semiconductor design as an actual physical process. In order to check the pattern generated in the actual pattern in advance and solve the problem, the semiconductor design process mainly uses the technique of predicting the result value for a given design by creating a model that expresses the actual physical process as it is. In general, Optical Proximity Correction (OPC)[4] models an optical system in which a photomask pattern is formed on a wafer surface or a process in which a photosensitive material reacts with light. In this process, in order to reduce the error between the model and the actual physical process, a test pattern is made into a wafer and then observed with a Critical Dimension Scanning Electronic Microscope (CD−SEM) to calibrate the error.

In the case of the latest semiconductor chip, it is often large enough to store more than trillions of polygons, and the test pattern consists of a combination of polygons. The accuracy of the model expressing the process is determined according to the type of test pattern used in the process of modeling each process, and it is very important to accurately select a test pattern that reflects the characteristics of the process to reduce cost and experiment time.

Optical Proximity Correction (OPC) is a technology that accurately patterns circuits designed on wafers by correcting distortions such as refraction caused by the characteristics of light in the photo process[5,6]. According to Moore's Law[7], the number of transistors integrated into a semiconductor doubles every two years. This increase in density can be achieved by reducing the minimum feature size, but the limits of optical resolution are reached as the feature size decreases. In order to overcome the limitation of optical resolution, an optical element having a large numerical aperture (high-NA, aberrations)[8,9], off-axis illumination (OAI)[10,11], a short wavelength light source (ArF, ArF-Immersion)[12,13], and a phase shift mask (PSM)[14] were used. OPC is a technology that improves the limitations of photolithography to overcome these optical limitations and improves the semiconductor process so that patterns are formed identically to the design. The pattern of the photomask may be modified so that a pattern having a small or large line width may be formed into a pattern having a desired size. However, other distortions, such as rounded corners, are difficult to correct with normal resizing methods. In this case, use a precomputed bias table to correct errors by moving edges or adding polygons to the pattern.

Since the early 2000s, model-based optical proximity correction (MB-OPC)[15,16] has been used rather than a rule-based method[17] using a bias table. MB-OPC has been used in the semiconductor industry to improve linewidth uniformity and pattern fidelity in photolithography. Calculate the pattern edge deviation in the design using an aerial image instead of a bias table to compensate for linewidth variations and distortion of patterns due to optical proximity effect. The MB-OPC process helps improve process windows as well as distortion of patterns, and also includes sub-resolution assist feature (SRAF)[18,19], auxiliary patterns and techniques such as hammerheads[20] and serifs[21].

Unlike rule−based OPC, MB−OPC is performed using a model, so problems arise when the modeling is inaccurate. Model errors are most often the result of not accurately modeling the lithography process. To improve this, resist characteristics such as Chemically Amplified Resist (CAR)[22] and Negative Tone Development (NTD)[23], mask scattering effect modeling such as Mask 3D (M3D)[24] and etch loading effect[25] were included in model elements.

MB−OPC includes basic data collection for model calibration[26] and verification[27], and OPC validation. In order to implement and verify the model, it is necessary to extract hundreds to thousands of design circuit pattern samples and measure the CD through SEM, and accurate calibration is possible with accurate measurement.

1.2. SEM inspection and design−based metrology

OPC is most often performed using software packages provided by electronic design automation (EDA) vendors. This software provides simulation−based verification of the physical behavior of illumination systems and projection optics. On the other hand, resist development, mask 3D effect, and etch loading effect are difficult to characterize with general simulation, so they are modeled and generalized using a huge amount of experimental data. In this process, in order to reduce the error between the model and the actual physical process, the test pattern is made into a wafer and observed with a Critical Dimension Scanning Electronic Microscope (CD−SEM)[28] to correct the error.

One of the most accurate tools for collecting this vast amount of data is the CD−SEM. CD−SEM is an important tool for characterizing nanoscale materials. By using electrons instead of photons to capture images, CD−SEM achieves sub−nanometer spatial resolution, revealing topological and compositional features not visible with conventional optical microscopes. It can measure the shape, size and roughness of patterns formed on semiconductor wafers.

A common way to measure the size of a pattern is to derive the critical dimension (CD)[29] between edges using a 1D pattern with line/space. This approach does not utilize the entire image, but only a portion of the image by limiting it to the measurement window. However, when dealing with complex 2D patterns, the size of the measurement window becomes very small, and measurements are often unreliable unless more time is spent. Until the early 2000s, acquiring and measuring images using SEM equipment was not an easy task. This is because the wafer had to be loaded into the equipment, and the operator had to manually navigate to the measurement location and acquire images manually[30]. Due to the shape of the various patterns, CD measurements are usually performed manually and take a long time. Additionally, the introduction of 193nm resists, which are much more sensitive to

SEM e−beam exposure[31], required greater attention to both the focusing and measurement steps to obtain reliable results.

As designs shrink and product development challenges increase, we need the ability to quickly measure large samples for accurate optical proximity correction (OPC). Since the late 2000s, a measurement technique using design image has been applied to measurement recipe setup and measurement automation, which is called Design Based Metrology[32] (DBM). Unlike previous image−based measurement techniques, DBM technology sets up a measurement recipe based on the design image and matches between the design image and the SEM image to perform the measurement. With this device, fully automatic CD measurement is possible by entering the design coordinates. Based on the superimposition of the design image and the SEM image, the algorithm can accurately recognize the measurement position and achieve high positioning accuracy. This device greatly reduces measurement time in the typical automatic CD measurement time range and improves reproducibility by allowing different structures to be defined for addressing and focusing prior to measurement. This new system enables more accurate OPC modeling and OPC verification of a variety of products that for various reasons cannot be measured[33].

As shown in Figure 1, pattern alignment in DBM is image processing that calculates the correspondence between the design layout of the measurement point and the SEM image, and the area with the highest score is selected as the measurement point[34−36]. Because the pattern shape of the circuit can be deformed during manufacturing, the score must be calculated taking into account the difference in shape that occurs during manufacturing.

For accurate pattern alignment, it is necessary to accurately extract the shape information of the pattern included in the SEM image[37−39], and if the quality of the SEM image is low, the pattern shape information cannot be extracted from the image and pattern alignment will fail

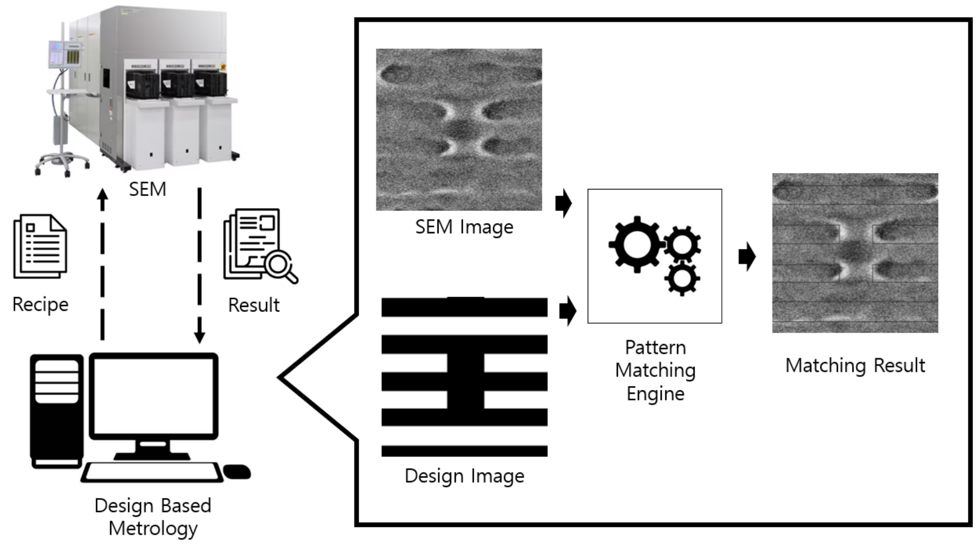


Figure. 1 Design based metrology system for pattern alignment

This pattern alignment is carried out based on image processing and the pattern alignment may fail depending on the quality of the SEM image. As the semiconductor design became smaller, various patterns such as EUVs and various patterning methods were applied, and the amount of additional work due to the failure of the pattern match increased rapidly.

1.3. A new method of lithography pattern alignment

A generative adversarial network[40] (GAN) is a class of machine learning frameworks designed by Ian Goodfellow and his colleagues in 2014. GAN can imitate any distribution of data through learning and create data in virtually every field. Two models, the Generator and the Discriminator, are created to perform adversarial training. The Generator receives a particular noise as input and generates an output in the same form as the data it wants to generate. The Discriminator identifies the output of the Generator and the training dataset of the data that it wants to generate. Through that process, the Generator is closer to the learning dataset, the Discriminator is more accurately identified, and the data that is closer to the real is created.

Since GAN first appeared, they have been gaining popularity in image processing, especially for transforming images from one domain to another. It is used in numerous fields such as image dataset generation[41], human face generation[42], realistic photo generation[43], cartoon characters[44], super−, resolution[45], text−to−

image translation[46], semantic-image-to-image photo translation[47], face aging[48], photo blending[49], video prediction[50] and image-to-image translation[51]etc.

Pix2Pix is an image-to-image translation model based on CCAN (Conditional GAN)[52], known as an effective way to transform an input image into an image of a specific domain. In Pix2Pix, the conditional input (x) is input to G to generate the image output G(x), and the input (x) and output G(x) are input together to Discriminator D(x). Target(y) exists in target domain, and target(y) enters Discriminator D(x) as input(x). Generator is a U-Net[53] with an encoder-decoder structure, and U-Net is a form in which symmetric skip connections are added to the encoder-decoder structure. Adding skip connections can solve the problem of the decoder not learning well. The Discriminator divides the image into N*N sized patches and classifies true/false for each patch and classifies it as 'true' if there are many true and 'false' if there are many false[54]. (Figure 2)

A new method of alignment the SEM image with the design image has been proposed[55]. First, use the unsorted data set to create training data pairs for training. Training data pairs are created using three-channel images, with the SEM image in channel 0 and the clipped design image in channel 1 to create one image. The training data is manually aligned and then clipped so that it is precisely aligned. Using the training data generated in this way, the generator is trained to convert the SEM image into the design image. The next step is to train the generator and discriminator to be able to accurately infer images. An inference step that converts SEM images into design images using a trained generator. Finally, align the inferred design image with the actual design image to obtain alignment coordinates. SEM images are precisely positioned using these coordinates.

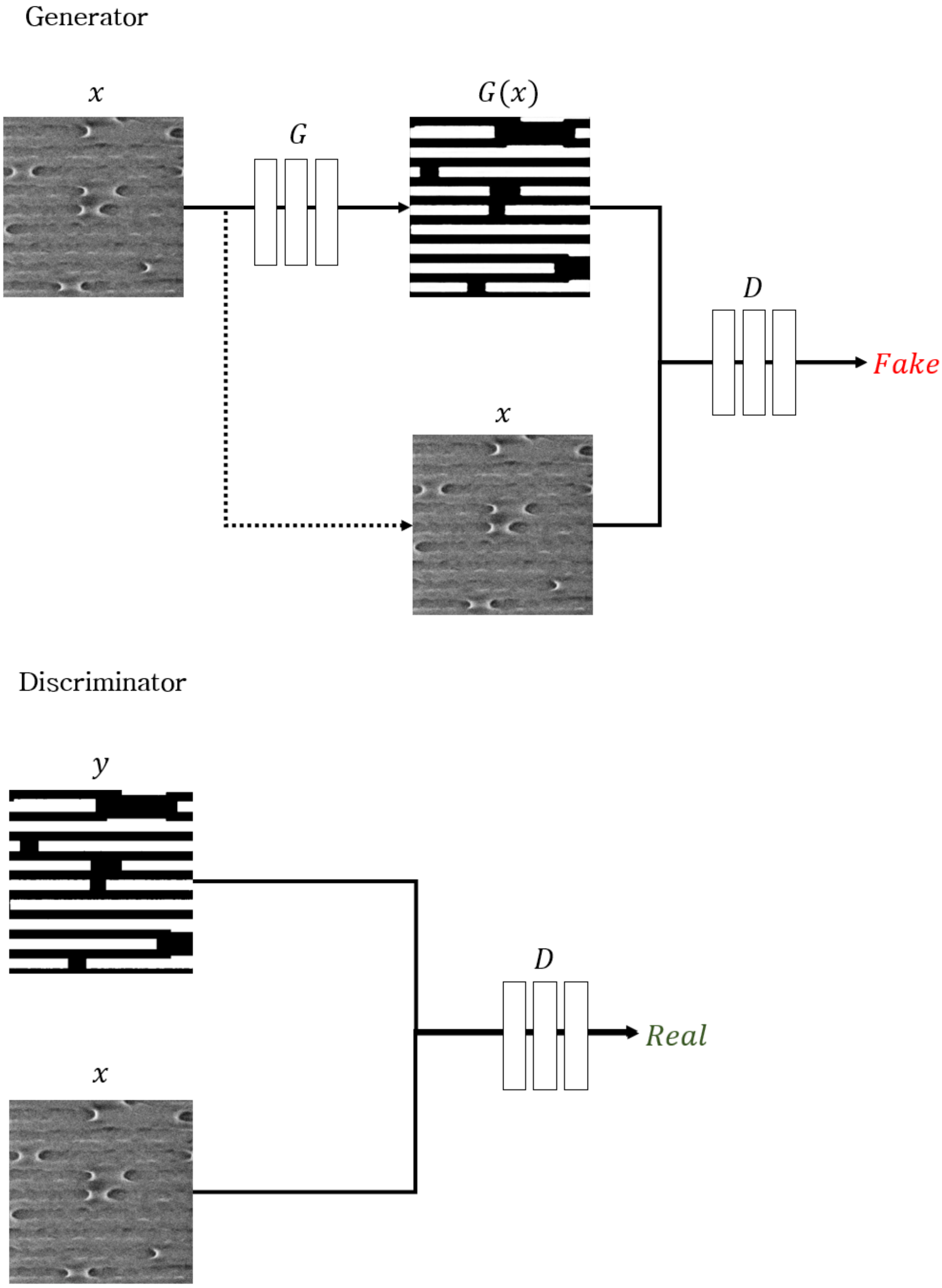


Figure. 2 Architecture of SEM image translation.

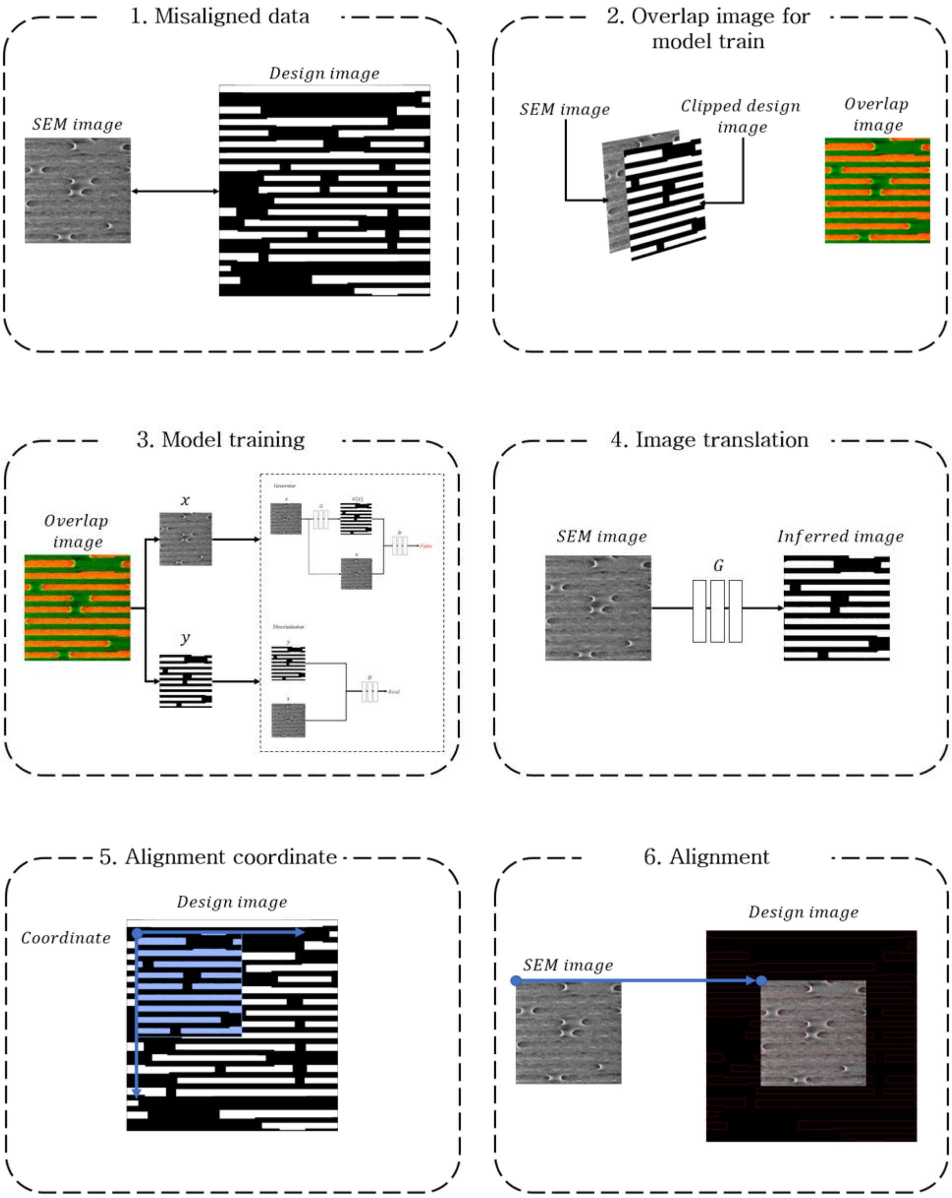


Figure. 3 Explanation of the proposed alignment.

We use Pix2Pix to effectively infer SEM images. Our approach models the process of removing noise and shape distortion from SEM images and can remove both noise and shape distortion from the expected design images. Also, the expected design layout has the same domain as the actual design images. Therefore, the match between the expected design images and the actual design images is much easier than the match between SEM image and design image due to noise-free, no morphological distortion, and simple intensity levels.

The most important thing in SEM-CAD alignment technology using Pix2Pix is to convert SEM images to design images. Ideally, we need to train the entire data set to accurately infer all images, but the time to train the model increases with the number of images. In some cases, it may take several hours or longer. This is because SEM-CAD alignment technology is proposed for accurate and fast alignment, so a lot of learning time does not fit the purpose. Training is required using minimal samples for immediate use. It is necessary to select an image representing the whole, use it as training data, and arrange it so that it is evenly distributed without bias. If these conditions are not met, model training will fail and the desired image quality will not be achieved.

This study proposes a method for effectively selecting for robust training of deep learning network for lithography pattern alignment.

Training, validation, and testing of deep learning usually requires a suitable data set to ensure robust use. To do this, we need to divide the dataset into different characteristics for each subset. For example, the design image is a binary image, which allows you to extract polygons from each image and classify the image using the extracted polygon. Using the classified image to confirm the effect of each image on the alignment accuracy. After determining the effect of each pattern, the optimal distance is inferred to select the sample set that can maximize alignment accuracy.

Chapter 2. Motivation

2.1. Relationship between accuracy and sample set

There are two main methods of training models in deep learning: supervised learning and unsupervised learning. Supervised learning is a method of training a model using data with correct answer labels and inferring correct answers from another data set based on the trained data set. In contrast, unsupervised learning is a method of clustering similar features in data with no correct answer labels and predicting the outcome on a new data set. Unsupervised models can help automate the entire flow but have serious problems. Unsupervised models cannot correctly distinguish between patterns and backgrounds for unknown (or un−trained) images and can have unexpected intervals between inferenced images and input design images.

Supervised learning is a method of machine learning algorithms that learns from labeled data sets. This means that if the correct answer is not properly learned, the model will not be able to correctly distinguish patterns and backgrounds from unknown (or untrained) images, and there may be unexpected gaps between the inferred images.

This is one example of SEM−CAD alignment using Pix2Pix. The case below is a very difficult case in which the so−called 1D type consisting of lines and spaces and the 2D type called T2T are mixed in the shape of the pattern. The total number of samples is 23, of which 5 are 1D type composed of Line/Space and 18 are 2D type composed of T2T. (Figure 4)

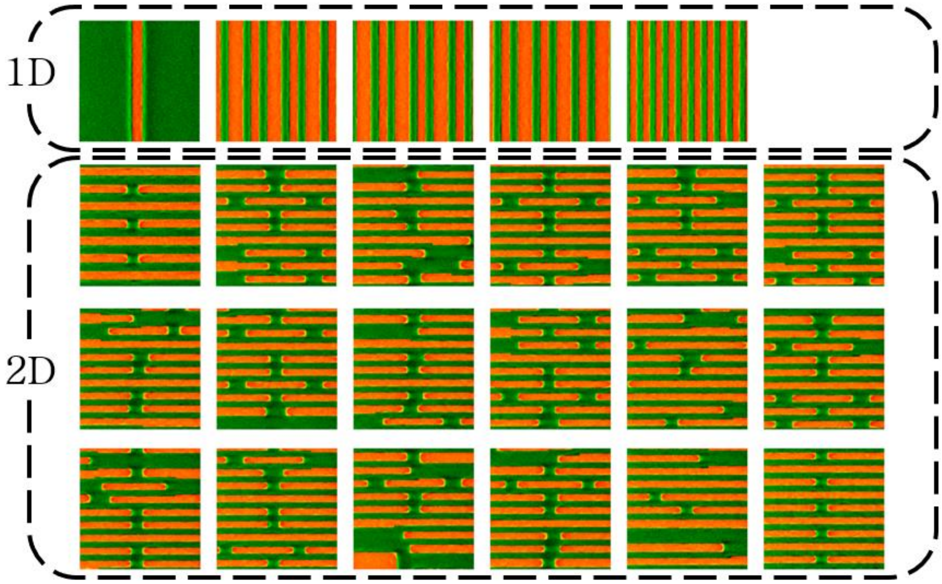


Figure. 4 Full training set of SEM-CAD alignment

As described above, training with all 23 images will give the most accurate alignment, but time and resource constraints make it difficult to use all images. So, we selected only a few samples and proceeded with training. In this case, only 7 samples were selected out of a total of 23 samples, and training and alignment were performed with only 7 samples. When using only 7 samples, we can see that the accuracy varies with the sample set. The sample set used in each case is summarized. (Figure 5,6)

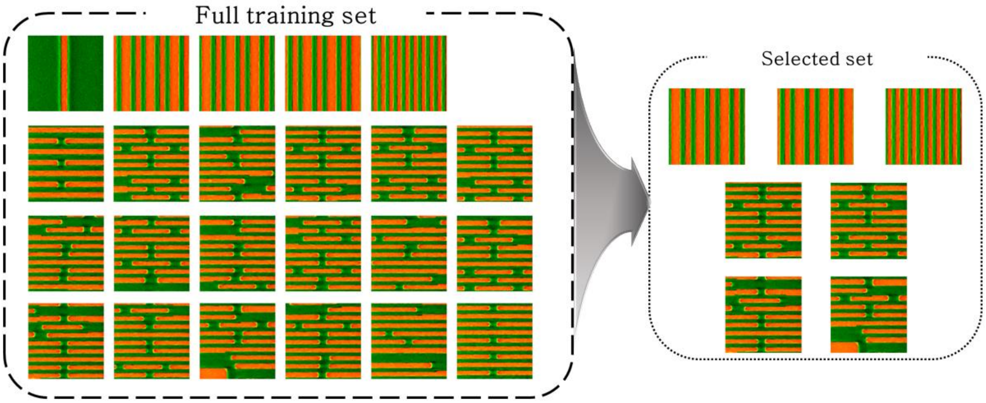


Figure. 5 Experiment 1 with only 7 samples selected

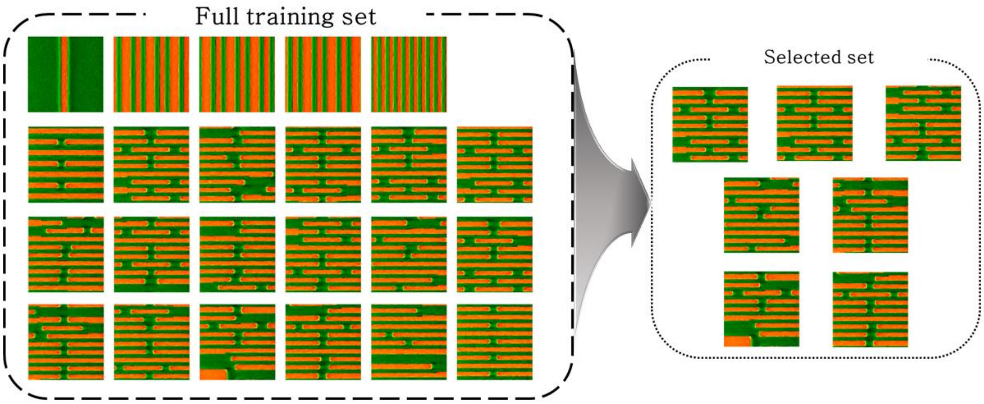


Figure. 6 Experiment 2 with only 7 samples selected

In both cases, the model was trained using 7 samples, but the accuracy is 97.24% for the experiment1 and 79.16% for the experiment2. So, why is there such a variation from sample to sample? Why is it not 100% successful? We analyzed the reasons why the two cases were not 100% successful in matching.

First, let's look at the sample set configuration of Experiment 1. The number of sample sets is 7 in total, 3 of which are the so-called 1D type composed of lines and spaces and 4 of the 2D type composed of T2T. When looking at only the 1D type consisting of lines and spaces, the difference from the full sample set is that it does not include the case where there is only one pattern called Iso. The entire sample set has an iso pattern, but the 3 1D patterns selected do not have an iso pattern. As a result, the model could not be accurately inferred for the iso patterns that were not included in the training, and due to this, the alignment accuracy was 97.24%, which did not reach 100%.

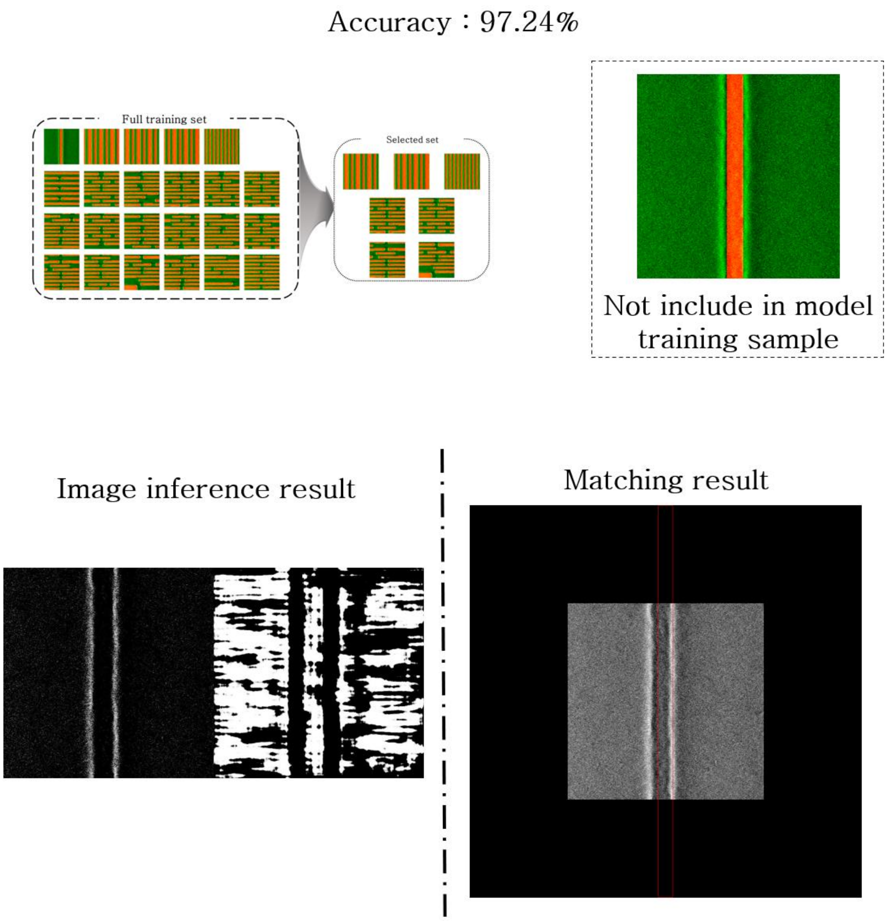


Figure. 7 SEM-CAD alignment failure case by ISO training sample

In Experiment 2, the composition of the sample set is related to the direction rather than the type of pattern. Among the entire patterns, the so-called 1D type composed of lines and spaces is in the vertical direction, and the 2D type composed of T2T is in the horizontal direction. All samples included in the training were 2D type and used only the horizontal direction and no vertical direction. This resulted in poor inference for the 1D type, resulting in an alignment accuracy of 79.16%.

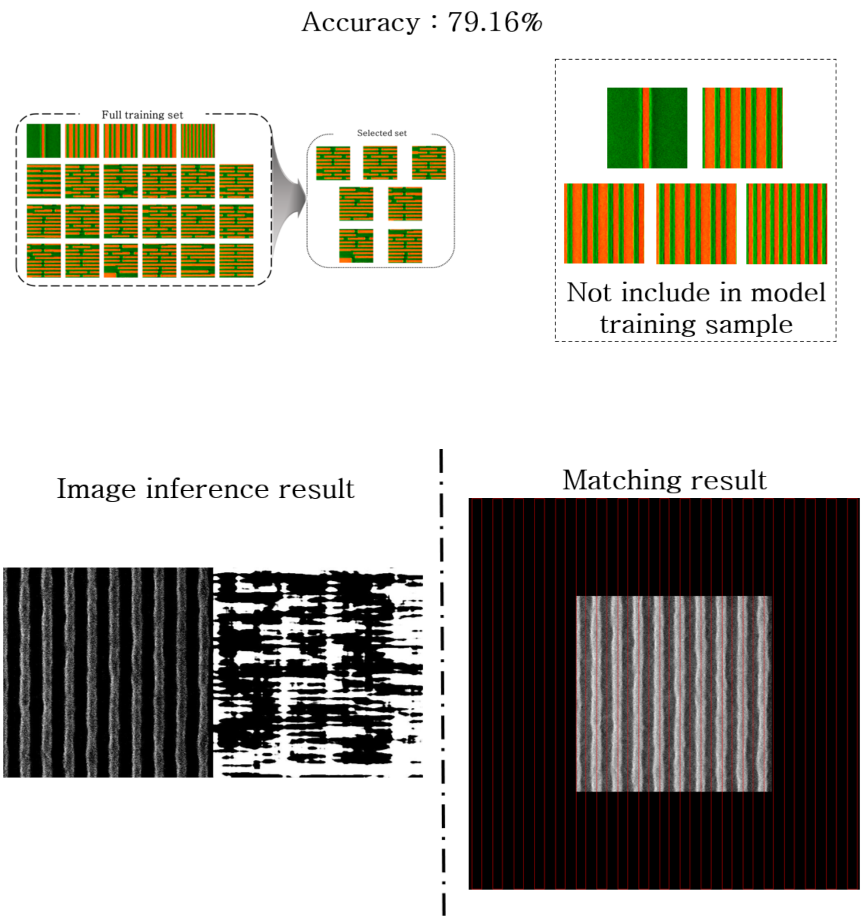


Figure. 8 SEM-CAD alignment failure case by direction of training sample

Accuracy varies from 79.16% to 97.24% depending on the type of pattern. Many factors affect the accuracy of the alignment, and it may be that the particular pattern is the problem, or the orientation of the pattern itself is the problem. Based on the above experiment, the samples were carefully reviewed and classified, and patterns were classified according to type and direction as follows. (Figure 9)

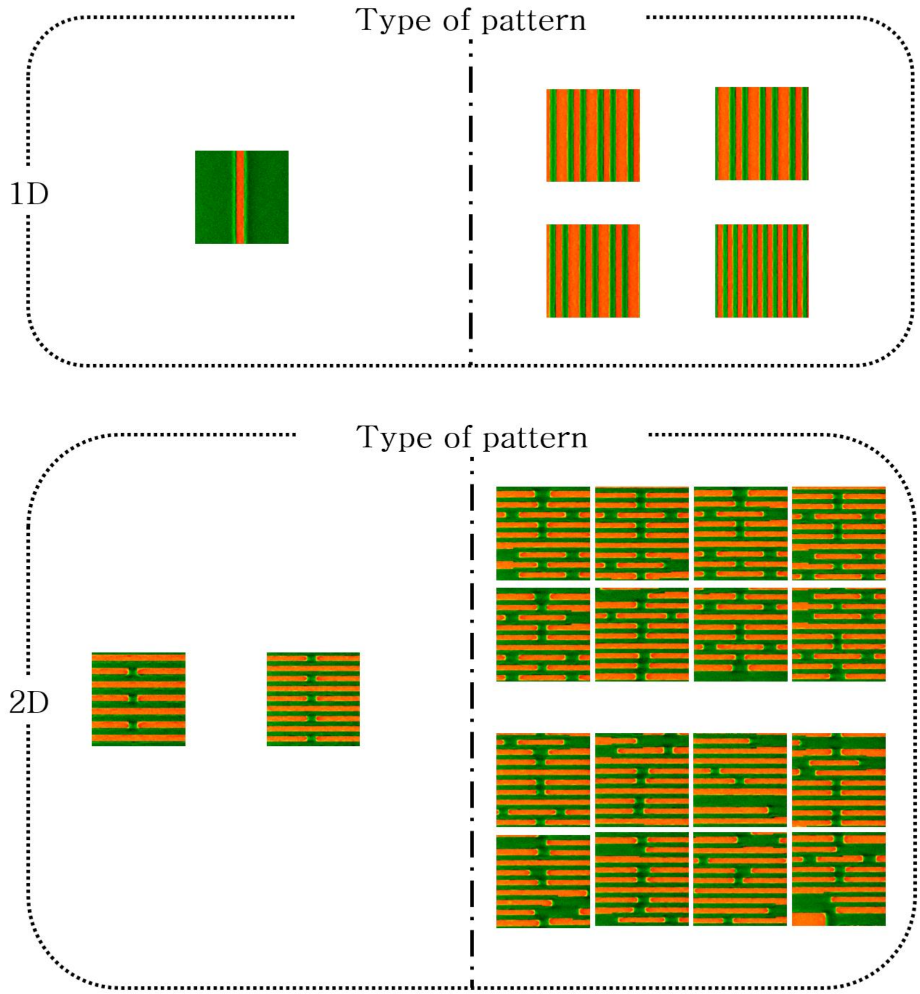


Figure. 9 Alignment sample selection according to pattern characteristics

A total of four samples were selected by selecting only one representative pattern among the classified samples, and training and alignment were performed with only the four samples. We were able to train a model that could achieve 100% concordance with only 4 samples. Through this experiment, it was found that it is more important to accurately analyze and correctly select a set of samples than to increase the number of samples in order to accurately train the model. If you choose the right samples, you'll be able to match more accurately with fewer samples than you can now.

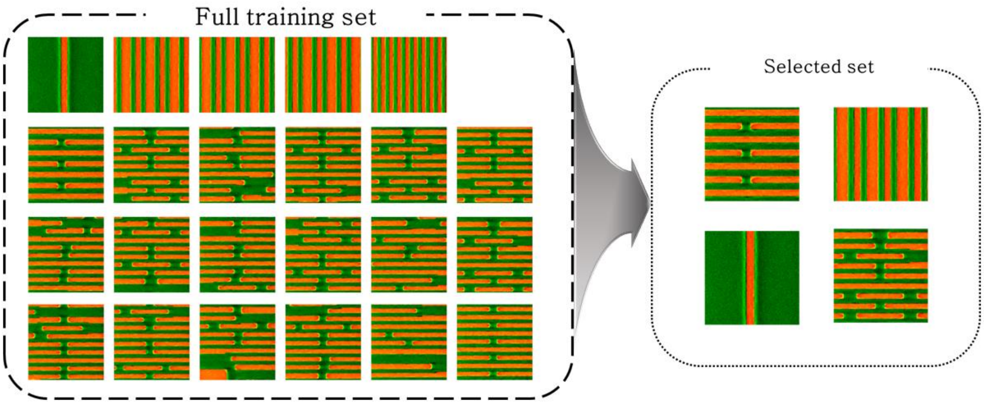


Figure. 10 SEM-CAD alignment success case by optimal training sample

Chapter 3. Experiments

Through several cases, it was found that it is important to select an appropriate pattern to train the model. In order to select an appropriate pattern, it is necessary to identify and classify the characteristics of the pattern. SEM images are noisy and have poor boundaries, making it difficult to extract features that can be used for classification. Classify images using design images instead of SEM images. As mentioned earlier, design images are made up of GDS files, and GDS files are made up of basic geometric elements such as rectangles and polygons. This geometrical information consists of 0s and 1s, and polygonal information can be easily extracted. We can extract the number of polygons, width of polygons, and spacing of polygons from a design image. Finally, the design image can be classified according to the desired criteria using the extracted polygon information. (Figure 11)

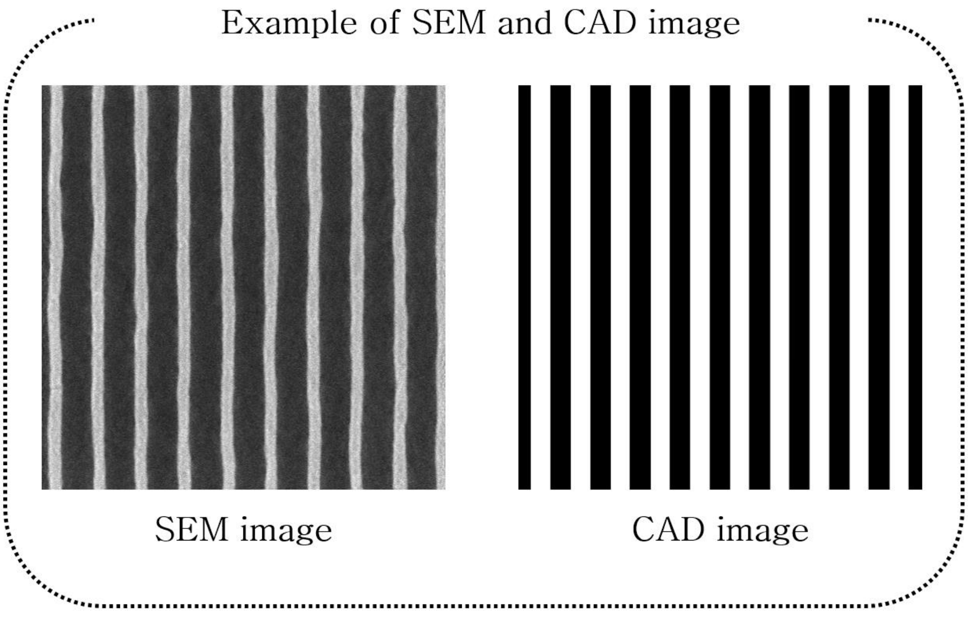


Figure. 11 Example of SEM and CAD image

In deep learning training, which converts SEM images into design images, it is important to select appropriate training images to improve alignment accuracy. Training images must be representative of the entire set, and not including the entire set can lead to serious alignment problems. To more clearly see the relationship between training images and alignment, we experimented with several cases. In order to confirm the relationship between learning images and alignment accuracy, polygon characteristics were extracted and patterns were separated, and the factors affecting alignment accuracy for each separated pattern were identified as follows.

1. Relationship between train pattern size and alignment result
2. Dense-to-iso or iso-to-dense pattern
3. The pattern shifted and direction

3.1. Relationship between train pattern size and alignment result

Figure 9 shows the inference results after training the model using one pattern. The size of the target polygon is 27 pixels, the spatial area that looks black in the SEM image is inferred as a spatial area of 27 pixels, and the photoresist (PR) next to the space that looks gray is drawn with a line of 27 pixels. When the model is trained using this image, the model will infer the photoresist (PR) next to the space as a 27-pixel polygon, and the actual inference result forms lines and spaces as predicted. To check the relationship between train pattern size and alignment accuracy, we test patterns of various sizes (2x, 3x, 4x) and compare the alignment results.

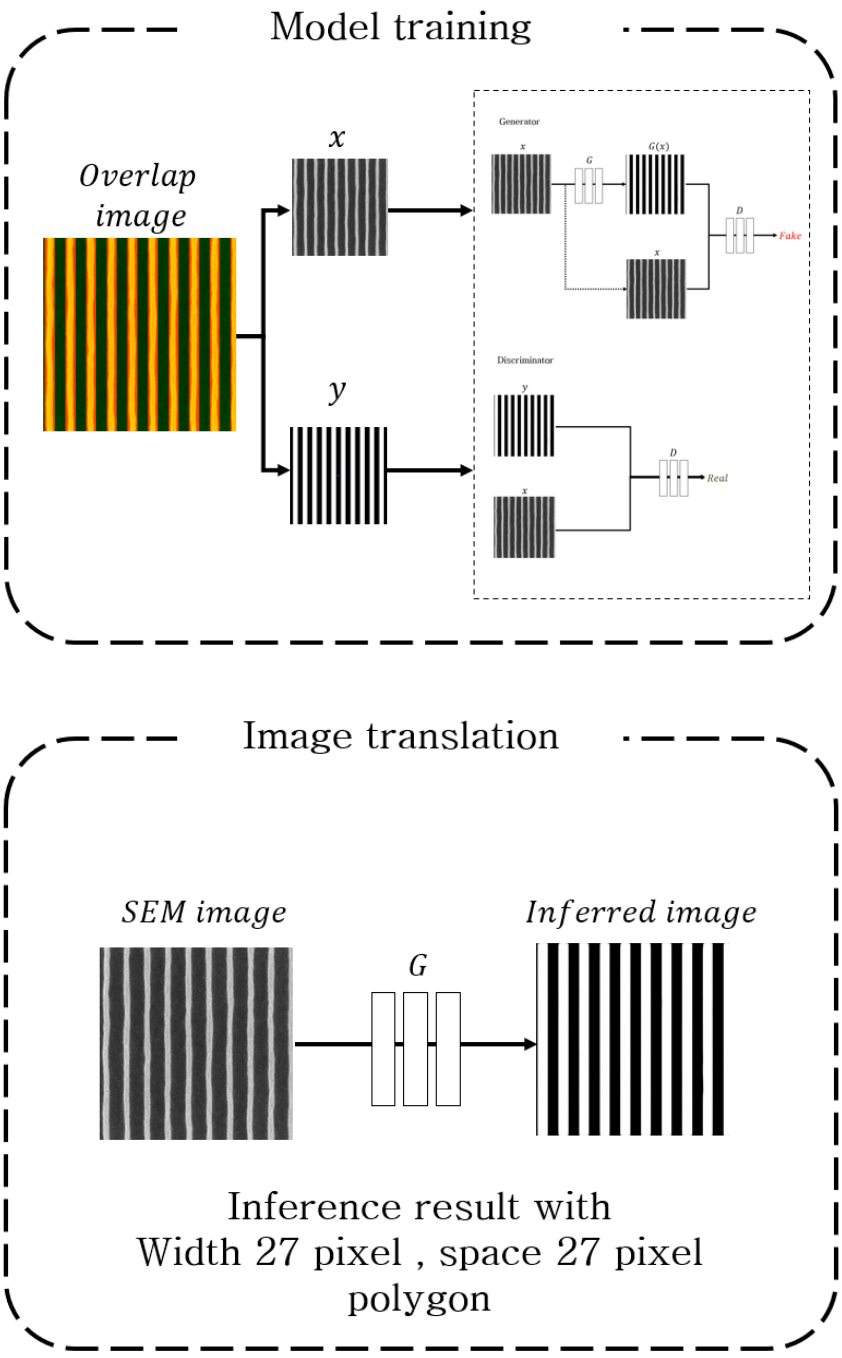
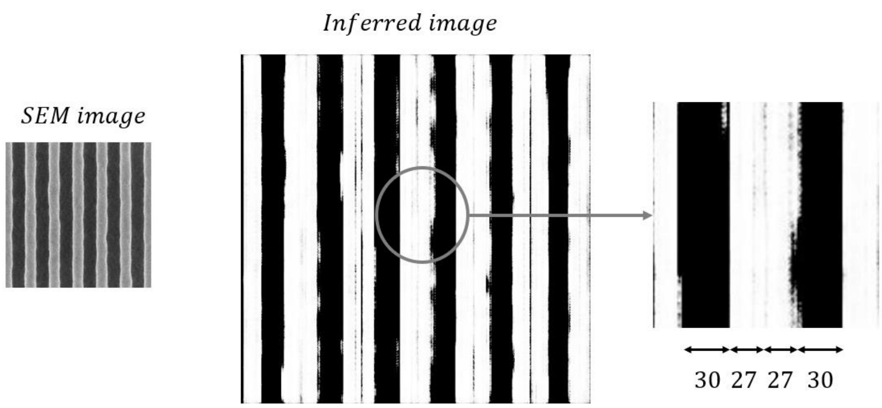


Figure. 12 Training result with w27s27 polygon

3.1.1 The result of inferring an image twice the size of the pattern learned by the model

The target pattern size for case1 is 54 pixels x 30 pixels. The size of the polygon of the target pattern is twice the size of the pattern used for learning (27pixel), while the size of the space is 30pixel, which is almost similar to the space used for learning (27pixel). This is the result of inferring the SEM image of case1 as a CAD image using the trained model. Looking at the inferred result, the size of the space remains unchanged because it is almost similar to the space of the image used for training. On the other hand, Photoresist next to space was inferred as a polygon of 27 pixels. However, since the size of the polygon is twice the size of the trained image, the PR next to the space is converted to a polygon of 54 pixels, so there is no problem with alignment.

Image translation



Alignment result

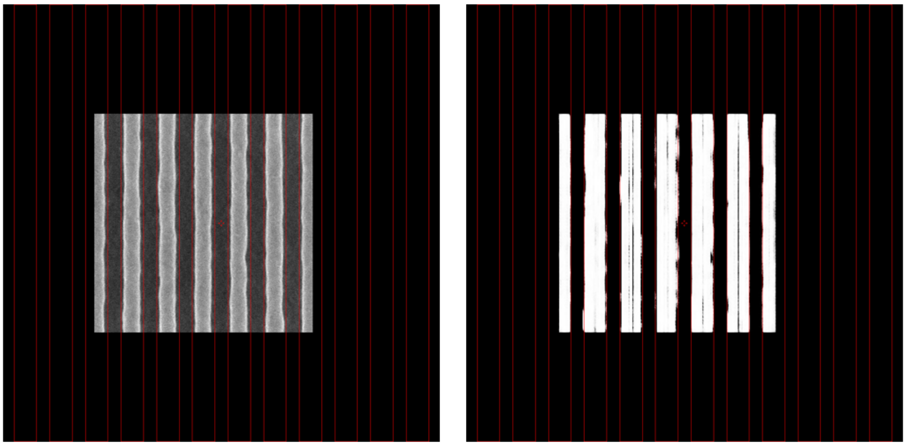


Figure. 13 Inference & alignment result with width 54 pixel polygon

3.1.2 The result of inferring an image three times the size of the pattern learned by the model

In this case, the size of the pattern is about 3 times the size of the learning pattern. The target pattern size for case2 is 79 pixels by 30 pixels. This is the result of inferring the SEM image of case2 to the CAD image using the trained model. Looking at the analogy result, since it is almost similar to the space of the image used for learning as in case 1, the size of the space remains the same, while the photoresist next to the space is inferred as a 27-pixel polygon. Since the size of the polygon is 3 times the size of the trained image, the PR next to space was converted to a polygon with 27 pixels on either side and a space in the middle. In this case, the polygon's centroid information is missing, potentially causing alignment issues.

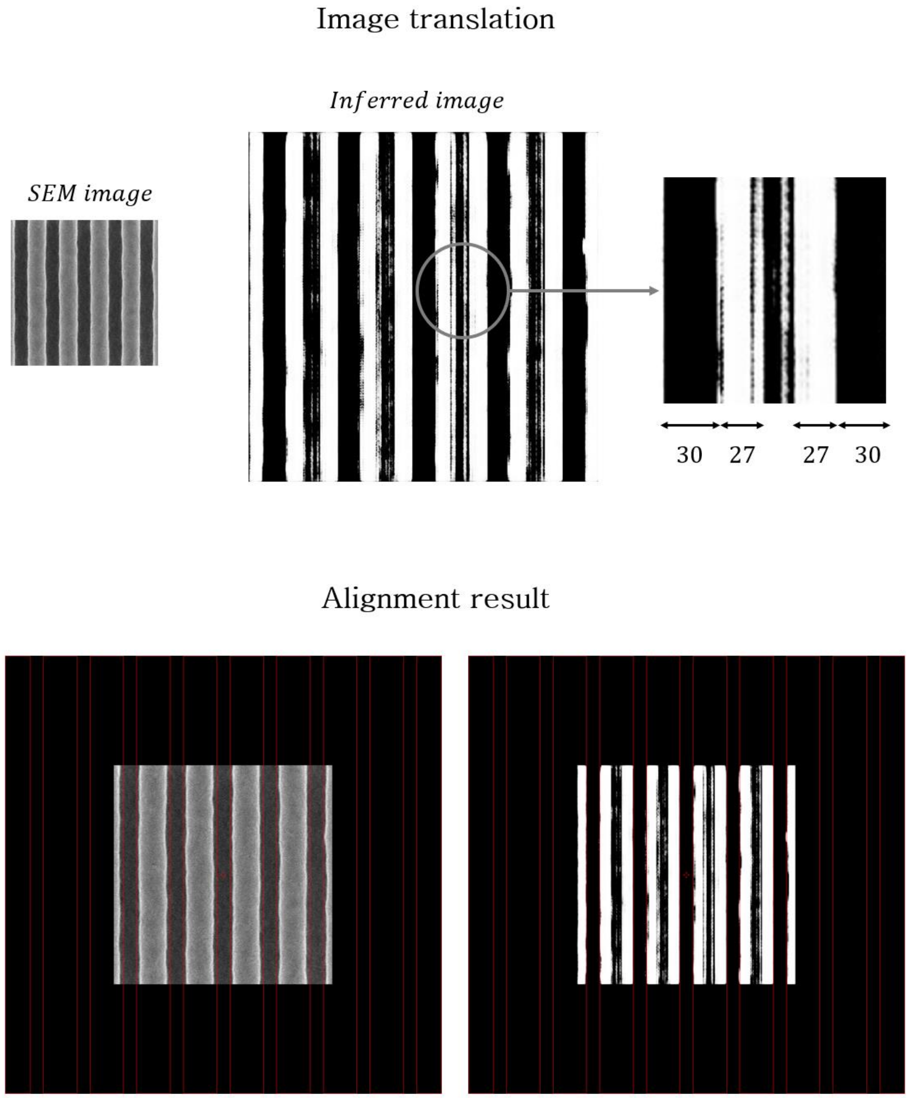
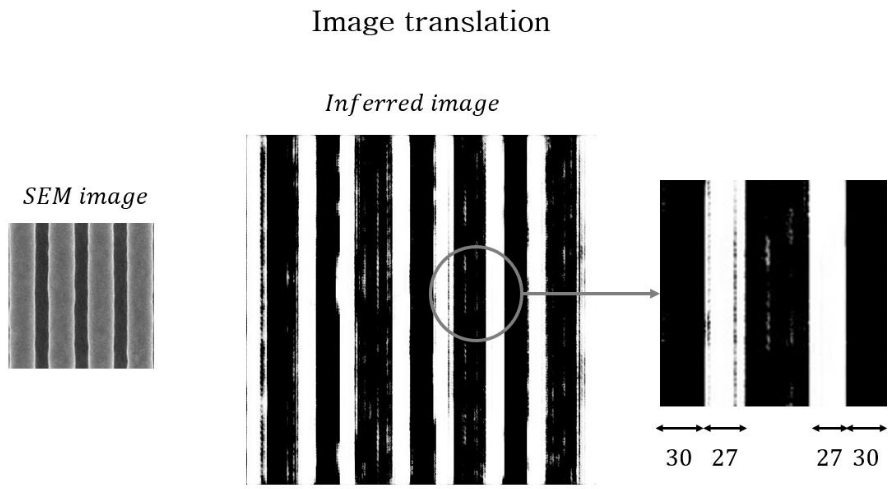


Figure. 14 Inference & alignment result with 79 pixel polygon

3.1.3 The result of inferring an image 4 times the size of the pattern learned by the model

For the last case, the target pattern size for case3 is 108px x 30px. The polygon size of the target pattern is 4 times the pattern used for learning (27 pixels), and the size of the space is 30 pixels, which is almost similar to the space used for learning (27 pixels). As a result of inferring the SEM image of case 3 as a CAD image using the trained model, the photoresist next to the space was inferred as a 27-pixel polygon. Similar to case2, since the size of the polygon is 4 times the size of the trained image, the PR next to the space was converted to a polygon with 27 pixels on either side and a space in the middle. The size of the space in the middle was created larger than the size of the space in the original target pattern, in which case alignment fails due to incorrect information.



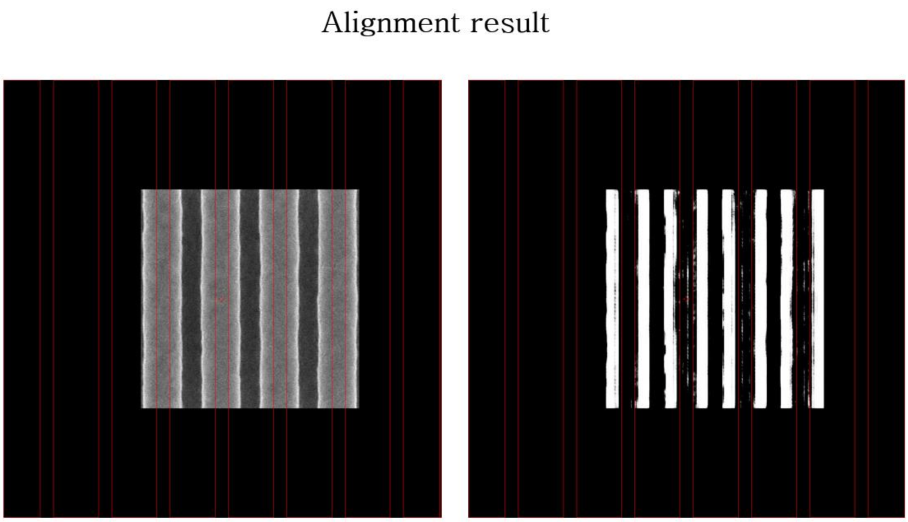


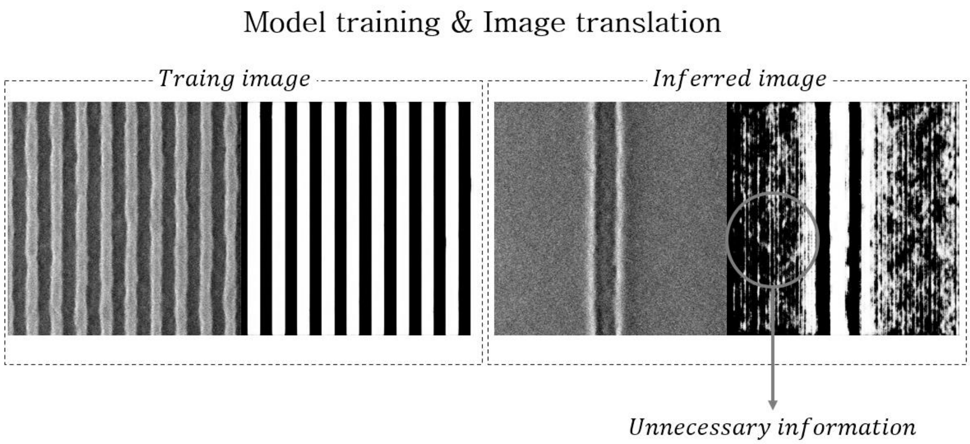
Figure. 15 Inference & alignment result with 108 pixel polygon

3.2. Dense−to−iso or iso−to−dense pattern relationship

Proximity effect is a change in line width of a feature (or shape of a 2D pattern) due to the proximity of another nearby feature. The simplest example of the optical proximity effect is the difference in printed line width between an isolated line and a line in a dense array of identical lines and spaces. This is called iso−dense bias.[56] OPC compensates for iso−dense bias by exposing a test pattern on the wafer and measuring the linewidth in various environments to determine the iso−dense bias. A relationship between these patterns and alignment was identified.

3.2.1 Dense–to–iso pattern relationship

In the first case, after learning a model using a dense pattern, inference was performed on the iso pattern to confirm the relationship between the type of pattern and alignment accuracy. In this case, contrary to previous experiments, the space between the photoresist is a polygon and the photoresist is a space. The line/space size of the dense pattern is 25 pixels by 28 pixels. After learning the model using this pattern, the inference result of the Iso pattern was confirmed. The inference around the iso pattern is relatively accurate, but it creates unnecessary images in the background area. These inappropriate inferences have the potential to affect alignment. (Figure 12)



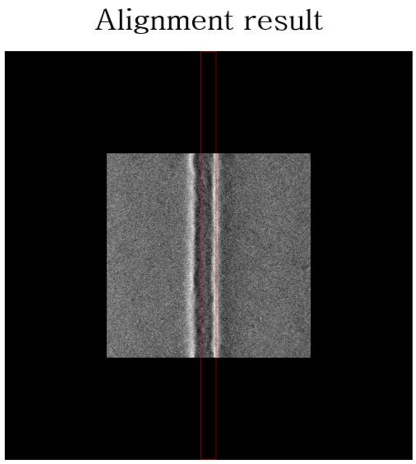


Figure. 16 Inference & alignment result with dense−to−iso pattern

3.2.2 Iso−to−dense pattern relationship

In the second case, the alignment accuracy for dense patterns was verified by training with iso patterns. The size of the pattern is 39 pixels, and after learning using the iso pattern, the reasoning result of the dense pattern was confirmed. Looking at the characteristics of the iso pattern used for learning, the rest of the area except the center is an empty space, confirming the tendency to follow the characteristics of these learning images as they are. The dense pattern inference results failed to form a pattern within a certain space, and the image was converted into a form with missing information. In this case, there is a very high probability that the alignment will fail because the pattern information is not delivered properly, and the alignment was also failed in the actual result. (Figure 13)

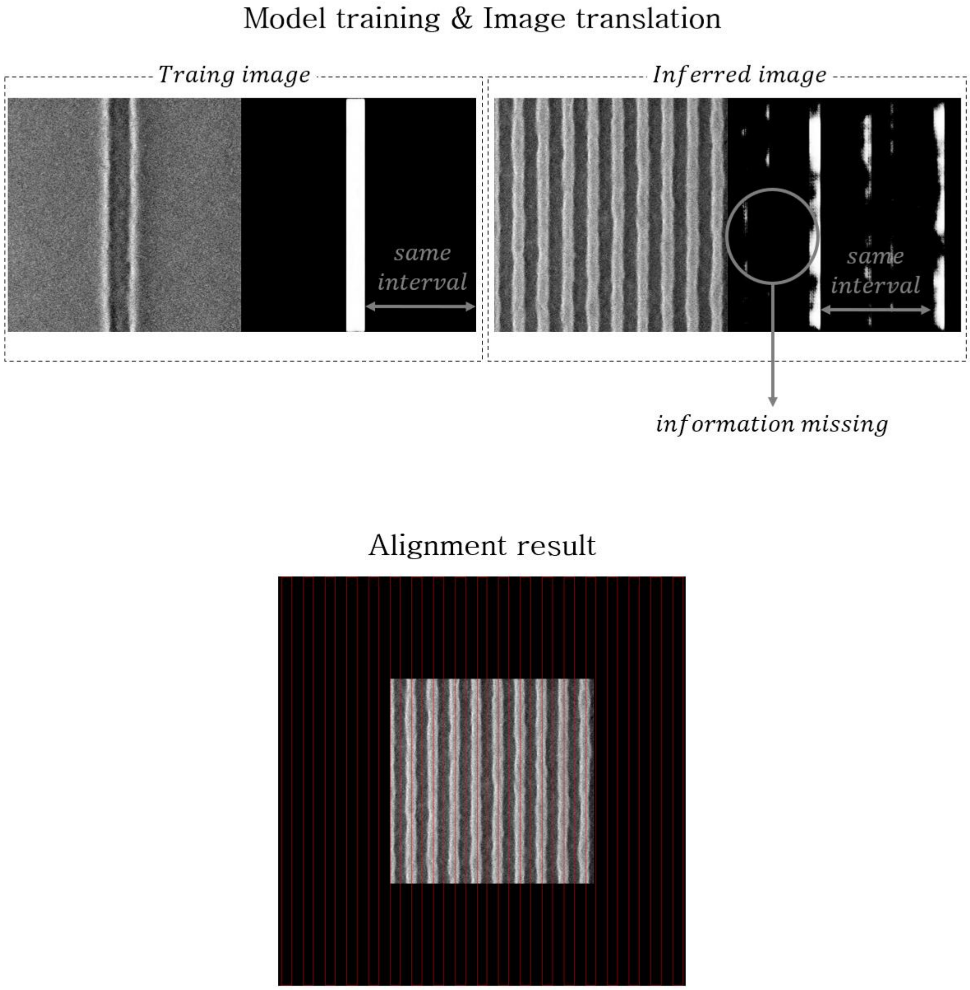
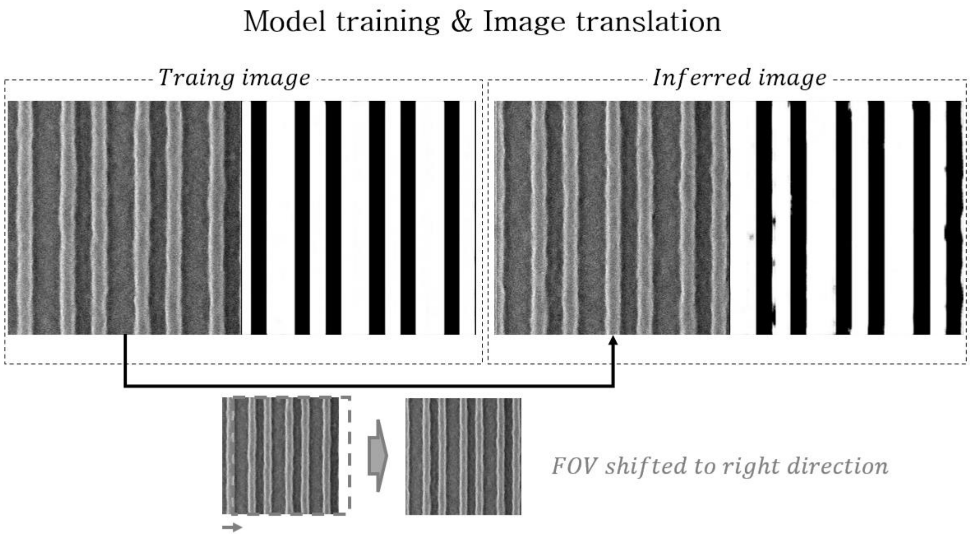
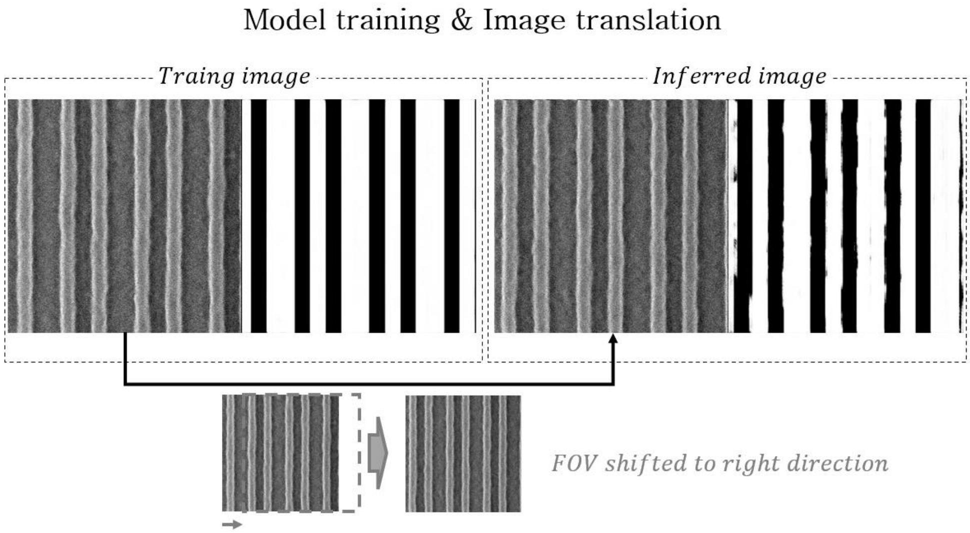


Figure. 17 Inference & alignment result with iso-to-dense pattern

3.3. The pattern shifted and direction

Ideally, the measured image should always be in the center, but movement in up, down, left and right directions often occurs due to instrumental errors. If the same image is moved and measured, it is checked whether a new training image needs to be added or the existing image can be used as it is. In the case below, it is an image composed of line/space, and the inferred image is confirmed while moving the image to the right. Because the FOV is fixed, if the pattern is out of alignment, the pattern seen on the screen may look different from the original image. A shift in which the shape of the original pattern remained within the screen by about 90% was found to be acceptable.





Alignment result

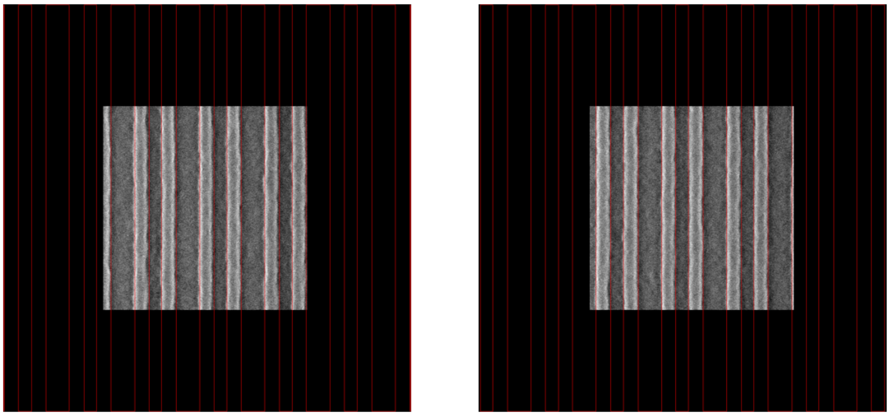


Figure. 18 Shifted pattern inference & alignment result

One factor to consider is the orientation of the measurement image. Since the instrument acquires images while scanning in the X direction during measurement, most of the measurement patterns exist in the vertical direction. However, in patterns such as T2T, the direction of patterns other than T2T is horizontal because the position to be measured is between the tips. That is, the pattern appears rotated. In this case, we checked the predictive ability of the model. As for the direction of the pattern, after training the model using the vertical pattern, the inference result for the pattern in the horizontal direction was confirmed. In this case, it was difficult to check the shape of the pattern because the reasoning was not done properly, and it was confirmed that the alignment failed.

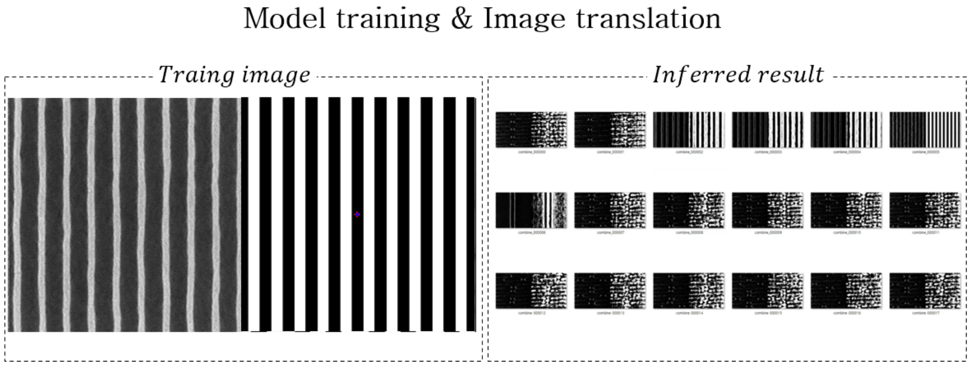


Figure. 19 Pattern direction and inference result

Through the above experiment, we were able to confirm the following facts. The most important thing when inference of a pattern from an SEM image to a CAD image is the size of the pattern. This is because inference is made only by the size of the pattern used for training. Training results are affected by both the width and space of the pattern. From the inference results for two different shapes (dense/iso) with similar widths, it was confirmed that additional patterns were formed or patterns were lost. If the pattern is shifted while maintaining the shape, there is no problem. But the direction of the pattern is very important for training. Since an appropriate training pattern must be selected according to the direction of the pattern, the training pattern must also be selected in consideration of the direction.

Chapter 4. Methodology and Result

Through previous experiments, we confirmed several cases that affect alignment. First, the size of the pattern directly affects the inference. It is very important to check the size of the pattern used for learning because we try to inference with the same size as the learned size. In addition, there are many factors to be checked in order to select a learning pattern, such as whether the pattern is dense/iso and direction. The process of selecting a pattern while checking these factors every time requires quite a bit of effort. We studied how to effectively select patterns while minimizing these efforts.

By analyzing failure cases, we learned how to construct train images for successful alignment. The goal of effective sample selection is to achieve 100% alignment with the fewest samples. In order to select the minimum sample from the entire sample set, the following two are essential.

1) All patterns in the sample must be classified based on their shape.

2) If the first clustering was done with a similar shape, then clustering according to the size of the pattern is required. (if the size of the pattern is more than 4 times larger than the image used in the train, it should be added to the train.)

Figure 16 shows the distribution of the line/space pattern of testcase. The width and space of the entire pattern were extracted using a test case with various types of line/space. Each blue dot in the picture represents one image. This graph contains information for a total of 1000 images.

In order to check the alignment accuracy for the learning pattern, after learning with one pattern, another pattern was inferred and aligned. It is necessary to establish a criterion to check the alignment accuracy by learning pattern. Alignment accuracy was verified by comparing the image shifts for each case based on the image shift values for the cases learned using all training datasets.

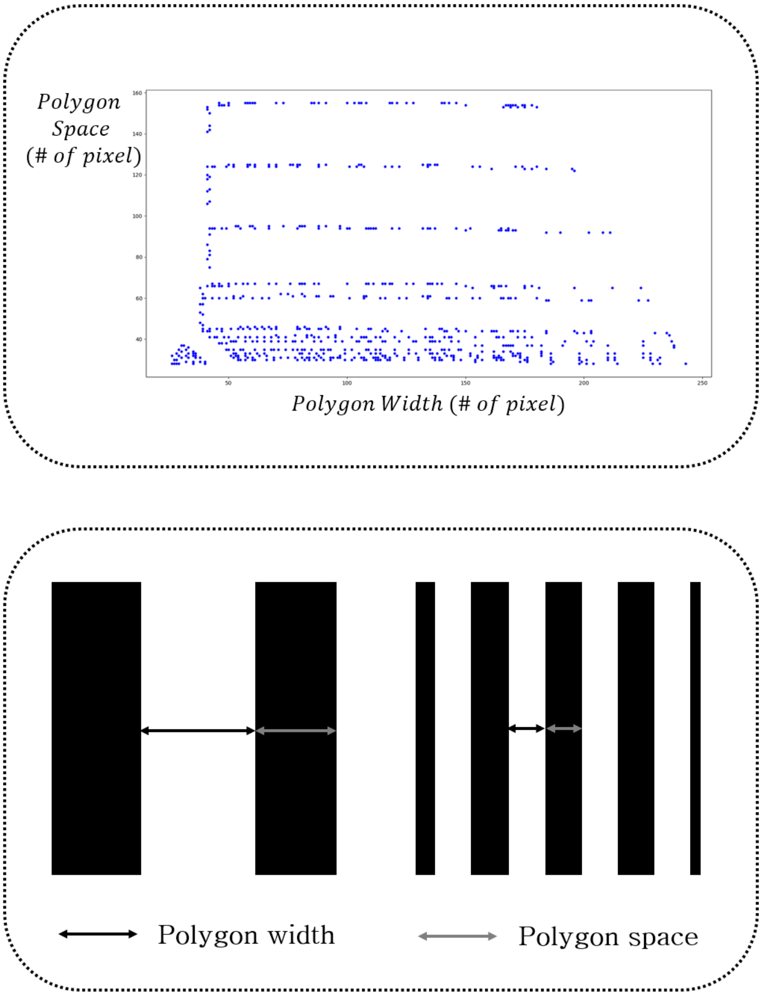


Figure. 20 Pattern distribution according to line/space

4.1. How to select the minimum sample that can cover the entire pattern

When you train by pattern, the trained model infers from the learned size. If so, how well can the alignment proceed if the alignment is performed with the model learned in this way? In order to check the alignment accuracy according to the pattern, after training the model with one pattern, alignment was performed on the remaining 999 patterns to check the alignment accuracy. Successfully aligned patterns are marked with blue dots and misaligned patterns are marked with red dots. Below is a map of alignment accuracy for different patterns.

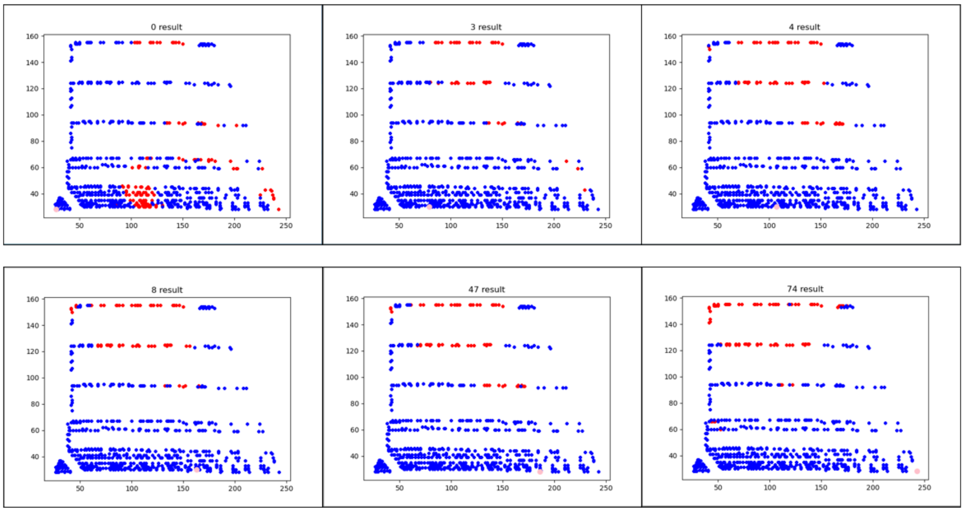


Figure. 21 A map that checks the alignment accuracy

We identified the alignment pass/fail relationship for all patterns. The most successful case of alignment with one pattern was when 930 patterns were successful. Based on this case, we studied how to successfully align the entire pattern while adding the minimum number of samples. Using the alignment pass/fail relationship for all patterns, select the cases in which the alignment of the most patterns can succeed, and filter the successful patterns. Next, select the case where the most patterns can be sorted out of the remaining patterns and exclude successful patterns. We figured that by repeating this process, we would be able to align the entire pattern with a minimal sample.

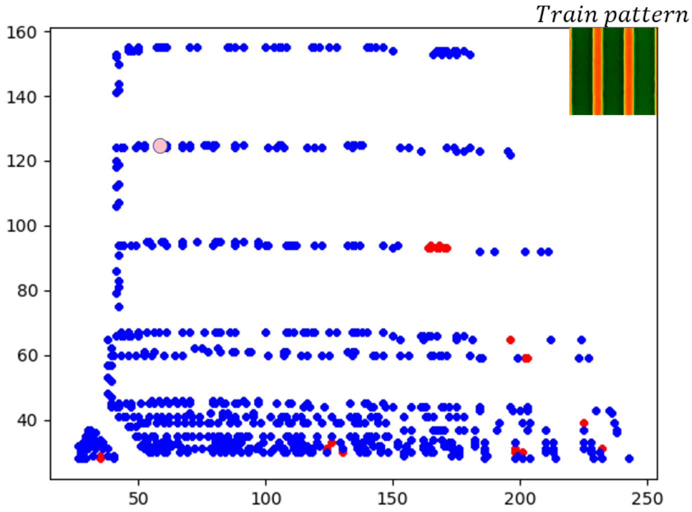


Figure. 22 A map for case where most of the patterns are successfully alignment

As a result of repeating this process, a total of three patterns were selected. After learning the model using three samples, alignment accuracy was confirmed by performing alignment on the entire pattern. Contrary to the expected results, alignment failed in some areas. Through this result, we learned that it is difficult to select a pattern that can cover the entire pattern simply by using only the pass/fail information of each pattern. So, how to select a pattern so that you can choose a sample that can cover the entire set with a minimum number of samples?

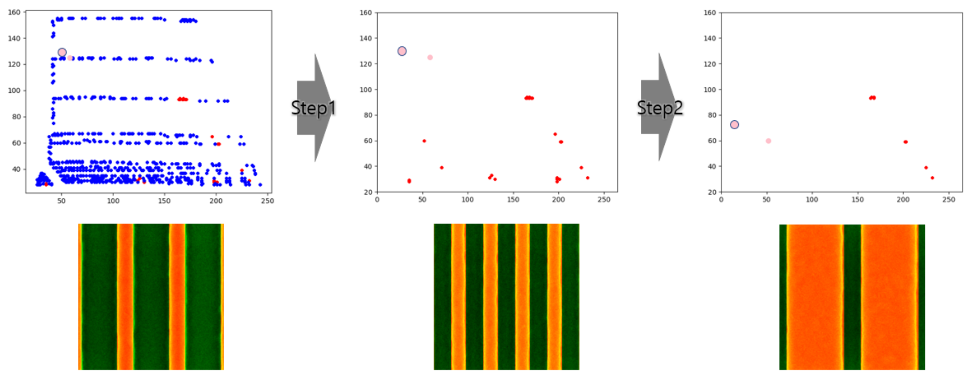


Figure. 23 Minimum pattern selection method using pass/fail information

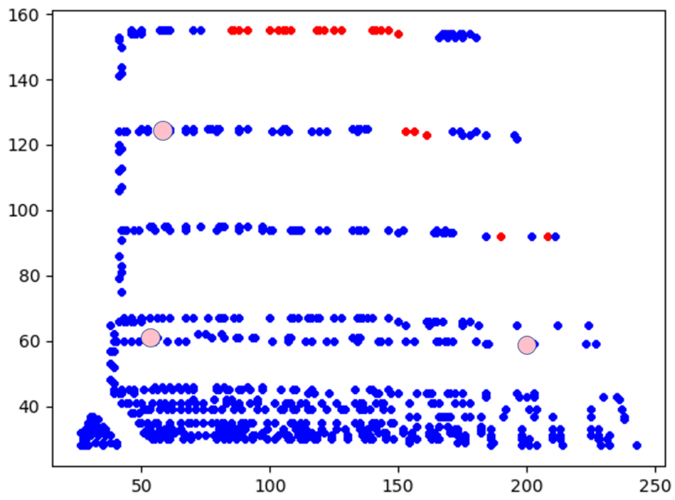


Figure. 24 Result using minimum pattern selection method using pass/fail information

4.2. Proposed method and improvement

The previous results showed the possibility of selecting a pattern that could cover the entire pattern, but it was not sufficient. Judging from the previous experience, it is very difficult to accurately identify the pattern covered by one pattern. When multiple patterns are used for learning, the result is different from simply adding the results of each pattern. In order to solve this problem and select a pattern more clearly, we want to select a pattern by designating a region. In addition, in order to improve the method of specifying the region, the concept of minimum effective distance was introduced.

4.2.1 Proposed method

The dense pattern, 27 pixels wide and 30 pixels apart, is the smallest pattern in the entire data set and is located in the lower left corner. After learning the model using this pattern, the alignment accuracy was verified by inferring the remaining 999 images using the learned model. Previous experiments have confirmed that there is a relationship between training pattern size and alignment accuracy. In this experiment, it was confirmed that the alignment failed a lot in the area 4 times the size of the pattern. In this way, the area where the alignment failed a lot by the model trained with one learning pattern was called a hidden region. The hidden region is not an area with an exact range, and it takes a process of finding areas where many patterns fail experimentally.

Figure. 25 A map that checks the alignment accuracy after learning with the W27S27 pattern.

The hidden region is determined by the size of the pattern. For example, the image below is a pattern with a width of 180 and a space of 155. This pattern has the largest width and space in the entire data set and is located in the upper right corner of the entire map. After training the model, this is a map that checks the alignment accuracy for the remaining patterns. Unlike the case of width27 and space27 above, it was confirmed that alignment failed for almost all patterns except for the pattern with space of 155 in the entire map. As the alignment accuracy differs depending on the type of learning pattern, it is necessary to check the alignment accuracy after training the model for all patterns. As a result of checking the alignment accuracy according to the pattern using different patterns, it was confirmed that they had different hidden areas as shown below.

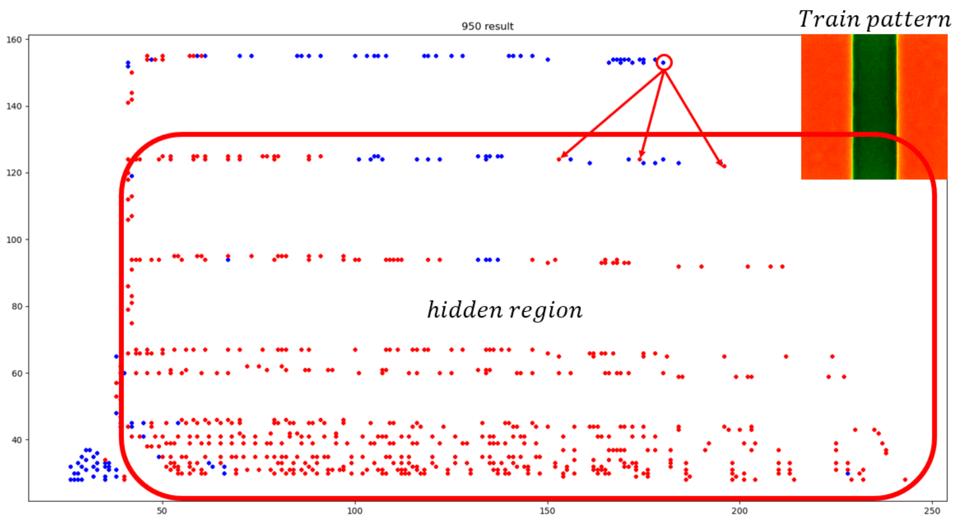


Figure. 26 A map that checks the alignment accuracy after learning with the W180S155 pattern.

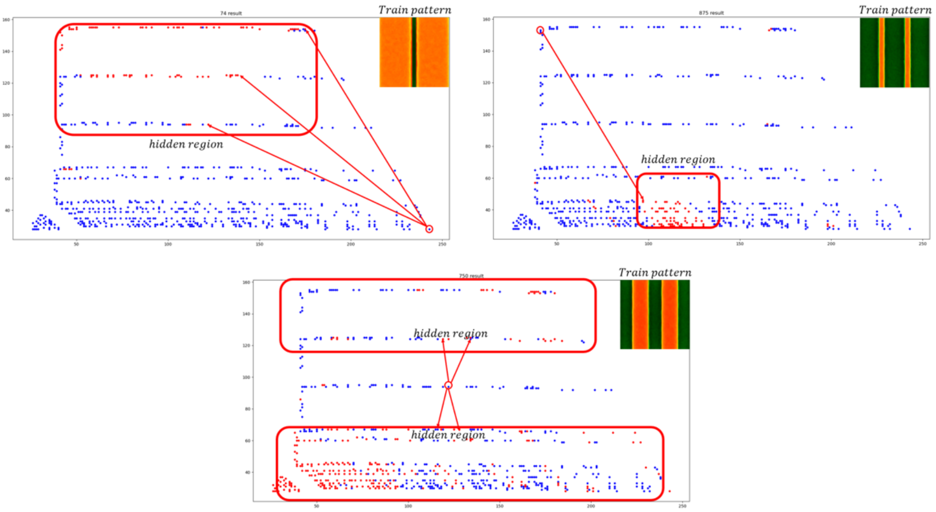


Figure. 27 A map that checks the alignment accuracy after learning with other patterns.

The hidden region is the area where alignment using each pattern fails. If the pattern is properly selected so that the hidden region is minimized, you can select a pattern that can cover the entire pattern. Based on the above idea, we figured the optimal case for selecting the entire pattern based on each hidden region. The left, right, top, bottom, and center patterns were selected from all patterns as shown below. Afterwards, the model was trained using only 5 patterns and the alignment accuracy was checked. As a result of checking, alignment was successful for all 1000 patterns. This is the result of confirming that 1000 images can be successfully aligned even if only 5 are used, if the pattern in the appropriate position is carefully selected using the information of the hidden region confirmed experimentally.

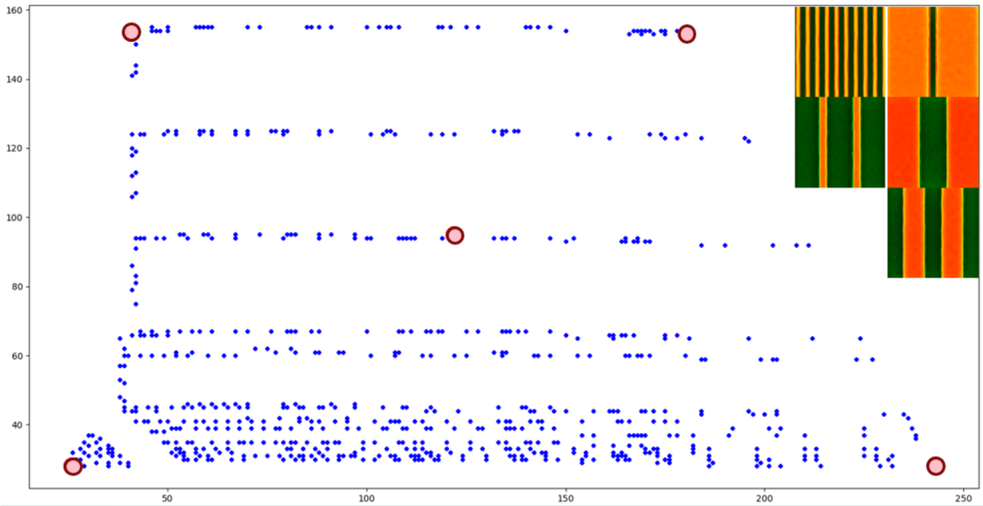


Figure. 28 Pattern alignment map using 5 patterns

4.2.2 Improvement using effective distance-based pattern extraction

Pattern selection using the hidden region can be determined by the individual's subjective point of view using an alignment map using a model learned with each pattern. If the individual's subjective judgment is included, it is difficult to increase the reliability of the test because the interpretation of the test result can be divided into various points of view. In order to clearly and quantitatively organize these experimental results, the concept of effective distance was introduced. After training the model on one pattern, perform alignment on the remaining patterns to see success/failure.

We calculated the distance from the training sample to the failing sample and defined the smallest value as the minimum valid distance. As with the hidden area seen earlier, the effective distance is specified differently for each pattern size. Therefore, the effective distance was checked for all samples. Unlike the hidden region, the smallest distance to the failed sample is used for the effective distance, so we can more intuitively figure out the relationship between training samples and alignment accuracy. In addition, since it can be expressed quantitatively, it is possible to exclude subjective elements and select an appropriate pattern through an accurate algorithm.

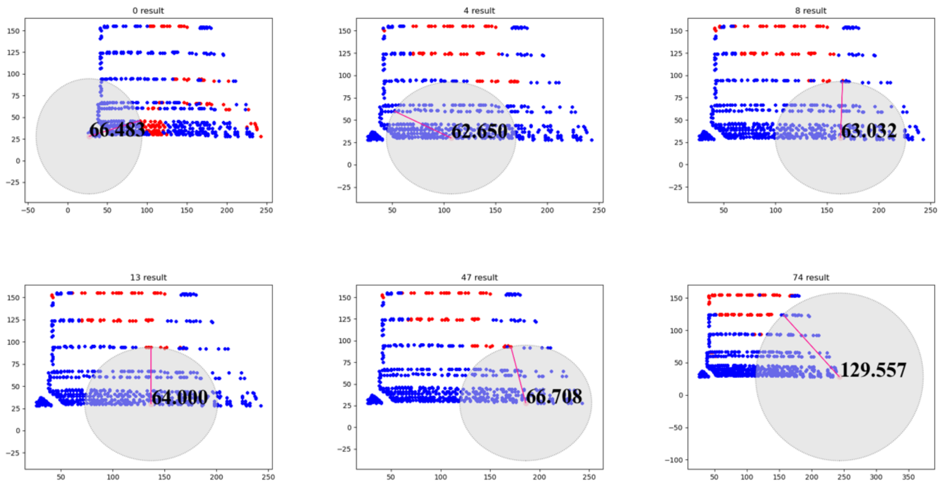
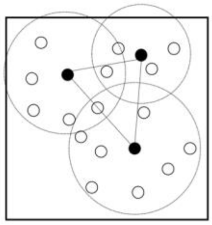


Figure. 29 Effective distance calculation for each pattern

To further clarify the results confirmed experimentally, we implemented an algorithm to determine the minimum pattern that can cover the entire area. This algorithm borrows the area coverage problem used in telecommunication networks. Area coverage problems monitor or cover the entire area of a specified network with the goal of leaving no points in the target area unreached by an observer. A point or target coverage problem is a special case of area coverage problem in which a limited number of mobile devices or points of interest are monitored or tracked by activating the required number of previously deployed sensor units instead of inspecting all points in a specific area.[57]

Since the area that can be covered is different for each pattern, the effective distance for all patterns is experimentally derived, and the algorithm is implemented to cover the entire area while minimizing the overlapping area using each distance. The implemented algorithm starts from the point with the largest effective distance, draws a circle with radius of effective distance, and removes points inside the circle. Then, the best sample is found by repeatedly removing the first operation using the point with the largest valid distance among the remaining points.



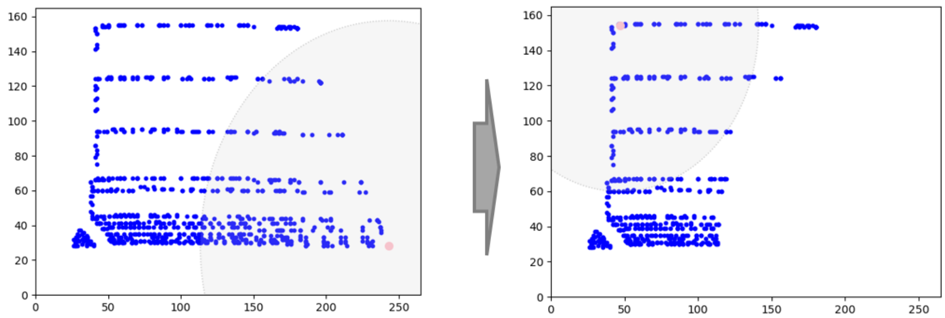
```
sort input_point_list according to their given radius

While input_point_list is not empty

        center of ith circle = input_point with largest given radius
        radius of ith circle = largest given radius
        remove input_point with largest radius
        find distance of all points to the center of ith circle
        if any point with distance < radius if ith circle
                remove any point with distance < radius if ith circle
                save center of ith circle
```

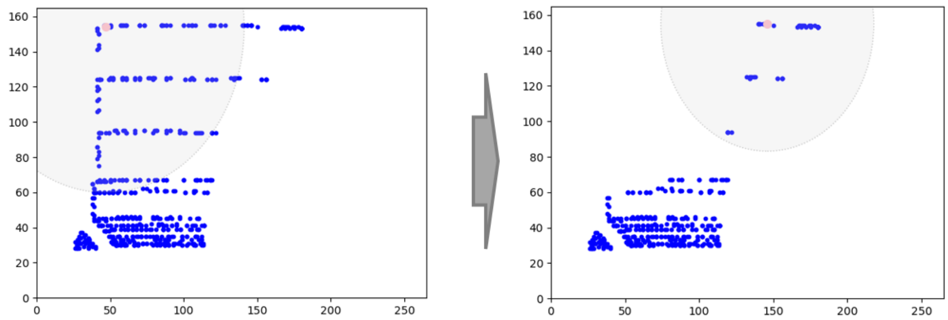
Figure. 30 Algorithm for minimum pattern selection

Pattern selection with effective distance



Iteration 1

Pattern selection with effective distance



Iteration 2

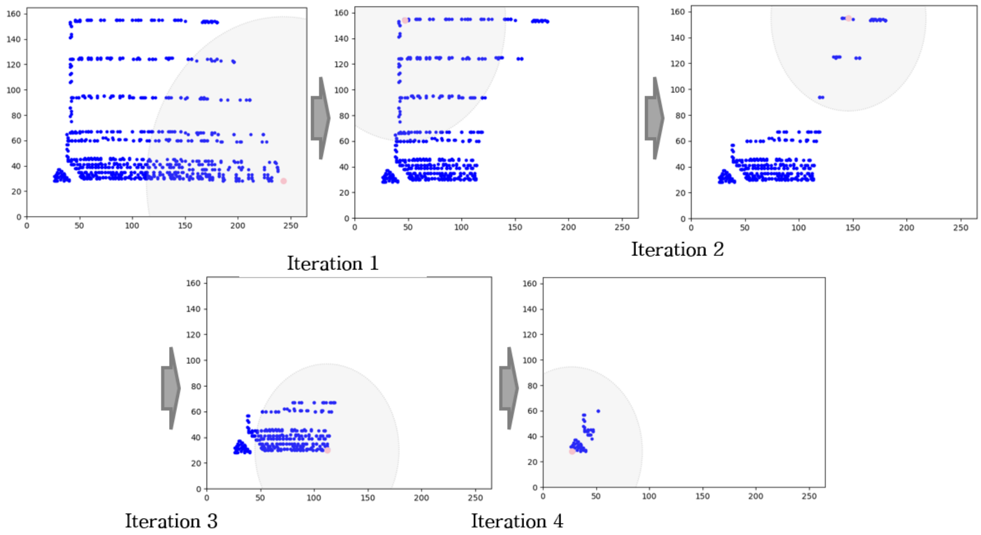


Figure. 31 Pattern selection flow

The first selected sample is the bottom right pattern with the largest effective distance. The size of the pattern is width 243 pixel, space 28 pixel, and the effective distance of this pattern is 129.5 pixel. Iteration 1 ends after excluding all patterns within the effective distance of this pattern.

Iteration 2, like iteration 1, selects the pattern that can cover the remaining patterns maximally, that is, the pattern with the largest effective distance. The pattern selected this time is the pattern on the top left, the size is width 42 pixel, space 150 pixel, and the effective distance is 86.1 pixel. By using this pattern, all patterns within the effective distance are excluded.

The result of extracting the sample is as follows. A total of 4 iterations were carried out, and 5 samples were selected, and the model was trained using the selected samples to check whether all samples were successfully aligned.

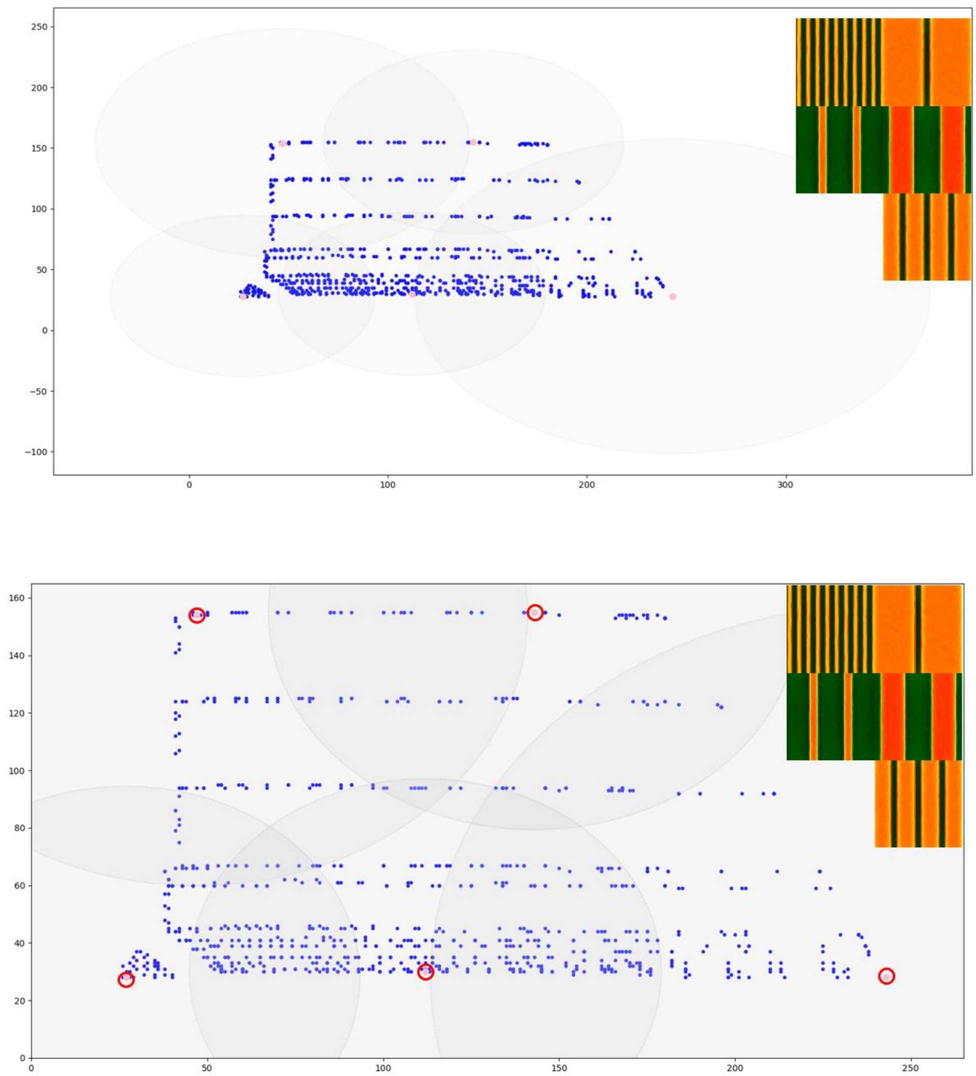


Fig. 32 Final result with effective distance and optimization algorithm

The final result using the effective distance and optimization algorithm were compared with the results of randomly selecting five patterns. A total of five randomized trials were conducted, and the average value was 85.3% and the standard deviation was 4.42, resulting in an average improvement of about 15% in accuracy.

It takes about 1 minute to find the effective distance for one pattern. In other words, it is a very large task that takes about 16 hours when checking all 1000 patterns However, since the map constructed in this way is created based on polygon information, it can be applied immediately without having to recheck similar polygon sets. Also, as the number of samples increases and the map becomes more sophisticated, the sample can be selected more reliably.

5 patterns training	Random selection Try1	Random selection Try2	Random selection Try3	Random selection Try4	Random selection Try5	Proposed method
Alignment Accuracy (%)	87.3	89.5	85.1	86.7	77.9	100

Table 1. Comparison results between random selection and the proposed method

Chapter 5. Conclusion

Optical Proximity Correction (OPC) is a technology that allows the design circuit to accurately pattern the design circuit on the wafer by correcting distortions such as the refraction caused by the characteristics of light in the photo process drawing complex electrical design circuits on the silicon wafer substrate during a semiconductor manufacturing process. In order to implement the exact pattern, hundreds to thousands of samples of the design circuit pattern must be extracted and the CD must be measured through the SEM, and the exact measurement enables accurate correction. The improvement of measurement ability is becoming increasingly important. Techniques that accurately recognize and measure the pattern you want will help improve the speed of measurement.

In this paper, we proposed a method to improve alignment by selecting a training set for deep learning model. The optimal sample was selected through effective distance-based pattern selection using optimization algorithms. In the past, the sample selection method was selected by the user's subjective method or a random method, but using the above method, a sample that can match the alignment 100% can be selected regardless of the user's personal experience Additionally, this method not only reduces extra work due to pattern inconsistency, but also reduces possible human errors in measurements. This is the first experimental study of how to select an appropriate training set to train an adversarial generative network model for lithographic pattern alignment. Based on the results of this research, we will conduct additional research to accurately measure OPC samples of more diverse shapes, which will contribute to the development of semiconductor processes.

Bibliography

1. L. Matti, V. Gerousis, M. Berekovic, P. Debacker, S. M. Y. Sherazi, D. Milojevic, R. Baert, J. Ryckaert, Ryoung-han Kim, Diederik Verkest, P. Raghavan, “Efficient place and route enablement of 5-tracks standard-cells through EUV compatible N5 ruleset”, in Proceedings Volume 10588, Design-Process-Technology Co-optimization for Manufacturability XII; 1058803 (2018)

2. "All About Calma's GDSII Stream File Format". www.artwork.com. Retrieved 2022-05-26.

3. https://www.samsungsds.com/kr/insights/eda_trend.html

4. Cobb, N.B, “Fast optical and process proximity correction algorithms for integrated circuit manufacturing”, in Spring (1998).

5. H H Hopkins, "On the diffraction theory of optical images", Proc. Royal Soc. Series A, vol. 217, no. 1131, pp. 408-432, 1953.

6. Cobb, N.B, Zakhor, A., “Fast sparse aerial image calculation for OPC”, in Proceedings Volume 2621, 15th Annual BACUS Symposium on Photomask Technology and Management (1995).

7. David C. Brock, “Understanding Moore’s law: four decades of innovation”, in Chemical Heritage Foundation (2006).

8. D.G.Flagello, A.E.Rosenbluth, C.Progler and J.Armitage, “Understanding high numerical aperture optical lithography”, in Microelectronic Engineering 17 105-108 Elsevier Science Publishers (1992)

9. BAHAA E. A. SALEH, M. C. TEICH, “Fundamentals of photonics”, in Wiley. p. 364-365. ISBN 978-0-471-83965-1. OCLC 22511619 (1991).

10. X. Chen and S. R. J. Brueck, “Experimental comparison of off-axis illumination and imaging interferometric lithography”, in Journal of Vacuum Science & Technology B: Microelectronics and Nanometer Structures Processing, Measurement, and Phenomena 17, 921 (1999)

11. A. Kroyan, D. Levenson, F. K. Tittel, “Coping with the impact of lens aberrations in the context of wavefront engineering” in Proceedings Volume 3334, Optical Microlithography XI; (1998)

12. R.Lindner, U.Stamm, R.Pätzel and D.Basting, “Advanced ArF excimer laser for 193 nm lithography”, in Microelectronic Engineering 41/42 (1998) 75-78

13. Chris A. Mack and Jeffrey D. Byers, “Exploring the capabilities of immersion lithography through simulation” in Proceedings Volume 5377, Optical Microlithography XVII; (2004)

14. Ronse, K. "Fundamental principles of phase shifting masks by Fourier optics: Theory and experimental verification", in Journal of Vacuum Science & Technology B: Microelectronics and Nanometer Structures. American Vacuum Society. 12 (2): 589–600 (1994).

15. John P. Stirniman, Michael L. Rieger, “Fast proximity correction with zone sampling ”, in Proceedings Volume 2197, Optical/Laser Microlithography VII; (1994)

16. Cobb, N.B, “Sum of Coherent Systems Decomposition by SVD”, September 21, 1995

17. Ji-Soong Park, Chul-Hong Park, Sang-Uhk Rhie, Yoo-Hyon Kim, Moon-Hyun Yoo, Jeong-Taek Kong, Hyung-Woo Kim, and Sun-Il Yoo,” An efficient rule-based OPC approach using a DRC tool for 0.18 /spl mu/m ASIC”, in Proceedings IEEE 2000 First International Symposium on Quality Electronic Design (Cat. No. PR00525)

18. J. Fung Chen, Tom Laidig, Kurt E. Wampler, and Roger Caldwell, “Optical proximity correction for intermediate-pitch features using sub-resolution scattering bars”, in Journal of Vacuum Science & Technology B: Microelectronics and Nanometer Structures Processing, Measurement, and Phenomena 15, 2426 (1997)

19. Xuelong Shi, Stephen Hsu, J. Fung Chen, Chungwei Michael Hsu, Robert John Socha, and Mircea V. Dusa, “Understanding the forbidden pitch phenomenon and assist feature placement”, in Proceedings Volume 4689, Metrology, Inspection, and Process Control for Microlithography XVI; (2002)

20. RenGuey Hsieh, Huitzu Lin, John C.H. Lin, Anthony Yen, Chue-San Yoo, Jia-Jing Wang, “Manufacturability of a 0.18-μm OPC technology”, in Proceedings Volume 3748, Photomask and X-Ray Mask Technology VI; (1999)

21. Alexander Starikov, “Use Of A Single Size Square Serif For Variable Print Bias Compensation In Microlithography: Method, Design, And Practice”, in Proceedings Volume 1088, Optical/Laser Microlithography II; (1989)

22. Xin Zheng, Jason Huang, Fred Kuo, Aram Kazarian, Fook Chin, and Yongfa Fan “Chemically amplified resist modeling in OPC”, in Proceedings Volume 7274, Optical Microlithography XXII; 727435 (2009)

23. Ao Chen, Y. Foong, Dongqing Zhang, Hongxin Zhang, Angeline Chung, D. Fryer, Yunfei Deng, Dmitry Medvedev, and Y. Granik, "Evaluation of compact models for negative-tone development layers at 20/14nm nodes", in Proceedings Volume 9426, Optical Microlithography XXVIII; 94261P (2015)

24. Finders J. and Hollink T., "Mask 3D effects: impact on Imaging and Placement", Proc. of SPIE, Vol. 7985, pgs. 79850l-1 79850l-23, 2011.

25. Won-Tai Ki, Seung-Hune Yang, Seong-Yong Moon, Seong-Woon Choi, Woo-Sung Han, and Jung-Min Sohn, "A new correction method for dry etch loading effect in photomask fabrication", in International Microprocesses and Nanotechnology Conference (IEEE Cat. No.00EX387) (2000)

26. Mohamed Bahnas, Mohamed Al-Imam, "OPC model calibration considerations for data variance", in Proceedings Volume 6924, Optical Microlithography XXI; 69243U (2008)

27. Cheol-Kyun Kim, Jae-Seung Choi, Byung-Ho Nam, DongGyu Yim, "Considerations of model-based OPC verification for sub-70nm memory device", in Proceedings Volume 6156, Design and Process Integration for Microelectronic Manufacturing IV; 61561D (2006)

28. I. Kawata, N. Hasegawa, "New Model CD-SEM for 45-nm Devices and Beyond", in Hitachi review (2007)

29. Francois Weisbuch, Jirka Schatz, Sylvio Mattick, Nivea Schuch, Thiago Figueiro, and Patrick Schiavone, "Investigating SEM-contour to CD-SEM matching", in Proceedings Volume 11611, Metrology, Inspection, and Process Control for Semiconductor Manufacturing XXXV; 116110Y (2021)

30. T. Ikegami, A. Yamaguchi, "Evolution and Future of Critical Dimension Measurement System for Semiconductor Processes", in Hitachi review (2011)

31. Bo Su, Guy Eytan, Andrew R. Romano, "193-nm photoresist shrinkage after electron-beam exposure", in Proceedings Volume 4344, Metrology, Inspection, and Process Control for Microlithography XV; (2001)

32. C. Tabery, L. Capodieci, C. Haidinyak, K. Shah, M. Threefoot, B. Choo, B. Singh, Y. Nehmadi, C. Ofek, O. Menadeva and A. Ben-Porath, "Design-based metrology: advanced automation for CD-SEM recipe generation," in Proceedings Volume 5752, Metrology, Inspection, and Process Control for Microlithography XIX; (2005).

33. Pietro Cantu, Gianfranco Capetti, Chiara Catarisano, Fabrizio D'Angelo, Elena Evangelista, Ermes Severgnini, Silvia Trovati, Mauro Vasconi, Takumichi Sutani, Stephan Wahl, Robert Steffen, "Evaluation of Hitachi CAD to CD-SEM metrology package for OPC model tuning and product devices OPC verification", in Proceedings Volume 5752, Metrology, Inspection, and Process Control for Microlithography XIX; (2005)

34. A. Sugiyama, Y. Toyoda, W. Nagatomo, T. Sutani, and R. Matsuoka, "A New Matching Engine Between Design Layout and SEM Image of Semiconductor Device," in Proceedings of the SPIE, Volume 5752, p. 546-558 (2005).

35. C. Tabery and L. Page, "Use of Design Pattern Layout for Automatic Metrology Recipe Generation," in Proceedings of the SPIE, Volume 5752, p. 1424-1434 (2005).

36. P. Cantu, G. Capetti, R. Steffen and T. Sutani, "Evaluation of Hitachi CAD to CD-SEM Metrology Package for OPC Model Tuning and Product Devices OPC Verification," in Proceedings of the SPIE, Volume 5752, p. 1341-1352 (2005).

37. E. Shauly, O. Menadeva, R. Drori, M. Cohen-Yasour, I. Rotstein, R. Peltinov, A. Bartov, S. Latinski, A. Siany, and M. Geshesl, "Accurate device simulations through CD-SEM-based edge-contour extraction," in Proceedings Volume 6922, Metrology, Inspection, and Process Control for Microlithography XXII; 69221L (2008)

38. K. Miura, M. Fujita, K. Nakamae, and H. Fujioka, "Pattern matching between a scanning electron microscopy exposed pattern image of large-scale integrated fine structures and computer-aided design layout data by using the relaxation method," in Journal of Vacuum Science & Technology B: Microelectronics and Nanometer Structures Processing, Measurement, and Phenomena 23, 3065 (2005)

39. T. Ikeda, T. Kotani, T. Sato, K. Ueno, and R. Matsuoka, "Pattern shape comparison methods using SEM image," in Proceedings of the SPIE, Volume 5038, p. 950-961 (2003).

40. Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville and Yoshua Bengio, "Generative adversarial nets," in Advances in Neural Information Processing Systems 27, Montreal, Quebec, Canada, pp. 2672-2680 (2014).

41. Alec Radford, Luke Metz, Soumith Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks", in conference paper at ICLR (2016)

42. Tero Karras, Timo Aila, Samuli Laine, Jaakko Lehtinen, “Progressive Growing of GANs for Improved Quality, Stability, and Variation”, in ICLR (2018)

43. Andrew Brock, Jeff Donahue, Karen Simonyan, “Large Scale GAN Training for High Fidelity Natural Image Synthesis”, in ICLR (2019)

44. Yanghua Jin, Jiakai Zhang, Minjun Li, Yingtao Tian, Huachun Zhu, Zhihao Fang, “Towards the Automatic Anime Characters Creation with Generative Adversarial Networks”, in IEEE International Conference on Advances in Electrical Engineering and Computer Applications (2022)

45. Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, Wenzhe Shi, “Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network”, in Computer Vision and Pattern Recognition (2017)

46. Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, Dimitris Metaxas,” StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks”, in ICCV (2017)

47. Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, Bryan Catanzaro, “High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs”, in Computer Vision and Pattern Recognition (2017)

48. Grigory Antipov, Moez Baccouche, Jean-Luc Dugelay, “Face aging with conditional generative adversarial networks”, in IEEE International Conference on Image Processing (ICIP) (2017)

49. Huikai Wu, Shuai Zheng, Junge Zhang, Kaiqi Huang, “GP-GAN: Towards Realistic High-Resolution Image Blending”, in Computer Vision and Pattern Recognition (2019)

50. Carl Vondrick, Hamed Pirsiavash, Antonio Torralba, “Generating Videos with Scene Dynamics”, in Computer Vision and Pattern Recognition (2016)

51. Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros, “Image-to-Image Translation with Conditional Adversarial Networks”, In CVPR (2017)

52. Mehdi Mirza, Simon Osindero, “Conditional Generative Adversarial Nets”, (2014)

53. Olaf Ronneberger, Philipp Fischer, Thomas Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation”, in MICCAI (2015)

54. Ugur Demir, Gozde Unal, “Patch-Based Image Inpainting with Generative Adversarial Networks”, in Computer Vision and Pattern Recognition (2018)

55. Yunhyoung Nam, Sungho Joo, Nohong Kwak, Kihyun Kim and Do-Nyun Kim, “Precise Pattern Alignment for Die-to-Database Inspection Based on the Generative Adversarial Network” in IEEE Transactions on Semiconductor Manufacturing, Volume: 35, (2022).

56. Chris A. Mack, “The Lithograpy Expert”, in Spring(1996)

57. Mohammad Mahfuzul Islam, Fariha Tasmin Jaigirdar, Mohammad Manzurul Islam, “Maximizing Network Interruption in Wireless Sensor Network: An Intruder’s Perspective”, in International Journal of Computer Networks and Communications (2015)

Abstract (Korean)

Critical Dimension SEM(CD-SEM)은 반도체 웨이퍼에 형성된 패턴의 모양, 크기 및 거칠기를 측정하는 전용 시스템이다. 설계가 축소되고 제품 개발 과제가 증가함에 따라 정확한 Optical Proximity Correction (OPC)를 위해 대량의 샘플을 신속하게 측정할 수 있는 기능이 필요하다. Design Based Metrology (DBM) 기술을 통해 설계 이미지를 사용하여 대량의 레시피를 빠르게 생성하고 측정 시간을 단축할 수 있었다. 그러나 디자인 이미지와 SEM 이미지 간의 정렬에는 여전히 많은 문제가 있었고, 이를 해결하기 위해 Generative Adversarial Network (GAN) 기술을 사용한 새로운 패턴 정렬 방법이 개발되었다. 본 논문에서는 디자인 패턴의 폴리곤 유형에 따라 학습 패턴을 분류하고 각 유형에 따른 정렬 효과를 확인하였다. 또한 훈련 세트와 정렬 정확도의 관계를 통해 모델 훈련을 위한 훈련 세트를 효과적으로 선택하는 방법을 연구하였다.