Deep Learning-Based Framework for Comprehensive Mask Optimization*

Bo-Yi Yu[1], Yong Zhong[1], Shao-Yun Fang[1] and Hung-Fei Kuo[2]

[1]Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan
[2]Graduate Institute of Automation and Control, National Taiwan University of Science and Technology, Taipei 106, Taiwan
{m10507413, m10607410, syfang, hfkuo}@mail.ntust.edu.tw

ABSTRACT

With the dramatically increase of design complexity and the advance of semiconductor technology nodes, huge difficulties appear during design for manufacturability with existing lithography solutions. Sub-resolution assist feature (SRAF) insertion and optical proximity correction (OPC) are both inevitable resolution enhancement techniques (RET) to maximize process window and ensure feature printability. Conventional model-based SRAF insertion and OPC methods are widely applied in industrial application but suffer from the extremely long runtime due to iterative optimization process. In this paper, we propose the first work developing a deep learning framework to simultaneously perform SRAF insertion and edge-based OPC. In addition, to make the optimized masks more reliable and convincing for industrial application, we employ a commercial lithography simulation tool to consider the quality of wafer image with various lithographic metrics. The effectiveness and efficiency of the proposed framework are demonstrated in experimental results, which also show the success of machine learning-based lithography optimization techniques for the current complex and large-scale circuit layouts.

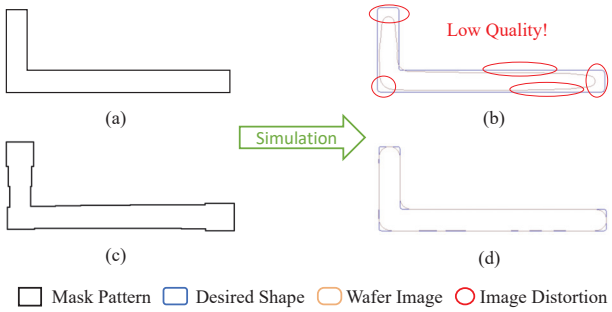

Figure 1: An example of masks with and without OPC and the corresponding wafer images. (a) Original mask layout polygon. (b) The corresponding simulated wafer image of (a) versus the target shape. (c) OPCed layout polygon. (d) The corresponding simulated wafer image of (c) versus the target shape.

1 INTRODUCTION

With the scaling down of feature sizes in integrated circuits (ICs) and the increasing of design complexity, semiconductor lithography techniques are facing serious challenges of wafer feature image distortion and process variation. For the past few years, many resolution enhancement techniques (RET) have been proposed to enhance pattern printability and fabrication yield [18]. Specifically speaking, in modern lithography process, optical proximity correction (OPC) and sub-resolution assist features (SRAF) play inevitable roles for wafer image optimization and process window maximization.

Optical proximity correction (OPC) is an important method to ensure the printability of wafer patterns which suffer from lithography diffraction and optical proximity effects. Figure 1 shows the simulated wafer images with and without executing OPC respectively. A layout polygon without OPC is shown in Figure 1(a), whose simulated wafer image is shown in Figure 1(b). As shown in Figure 1(b), the image contour is distorted in many sites and may become lithography hotspots on the wafer [2]. In contrast, the layout polygon with OPC (OPCed layout polygon) and the corresponding simulated image have higher lithography quality with less image distortion, as shown in Figures 1(c) and 1(d). In addition to wafer image distortion, process variation is also a main challenge in semiconductor manufacturing that can cause serious impacts on circuit yields. Process window (PW) is an important metric that quantifies the robustness of a manufacturing process considering process variations. PW defines the limits of variation parameters such as dose and focus, within which sufficient pattern printability and yield can be guaranteed. To tackle the issue of process variation, sub-resolution assist feature (SRAF) insertion has been proposed and shown its promising effectiveness of PW maximization [5, 15, 17, 19]. SRAFs are inserted to improve the depth of focus of isolated features without printing themselves on the wafer. Thus, the wafer patterns will be more robust to the process variations. However, with model-based SRAF insertion, the computational complexity of lithography simulation becomes even higher and thus inefficient.

Recently, some machine learning-based mask optimization methods have been proposed [6, 9, 11, 12, 20]. The major objective of these works is to efficiently get the initial OPCed mask patterns with machine learning models and reduce the optimization iterations, which contributes to reduce the overall runtime of model-based OPC tool. However, the existing machine learning-based mask optimization methods either use pixel-based mask optimization or lack for simultaneously SRAF insertion and OPC scheme, which respectively indicates their approaches fall short of mask manufacturability and sufficient wafer image quality as well as the ability to against process variation. In this paper, we propose the first, to the best of our knowledge, deep learning framework that simultaneously performs SRAF insertion and edge based-OPC for comprehensive mask optimization. To make the optimized masks more reliable and convincing for industrial application, we employ a commercial lithography simulation tool to consider the quality of wafer image with various lithographic metrics. Our experimental results show that our framework not only can generate the optimized masks with high manufacturability in extremely fast computational time, but the performance of ability to against process variation and the quality of wafer image are very competitive when it compare with commercial model-based OPC tool.

The rest of this paper is organized as follows: We introduce the previous works in Section 2. Section 3 gives some terminologies and the problem formulation. In Section 4, we detail each step in the framework for SRAF insertion and OPC. Experimental results are presented in Section 5. Finally, we conclude our work in Section 6.

*This work was partially supported by Synopsys and MOST of Taiwan under Grant No. MOST 107-2636-E-011-002.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ASPDAC '19, January 21–24, 2019, Tokyo, Japan
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6007-4/19/01…$15.00
https://doi.org/10.1145/3287624.3288749

2 PREVIOUS WORKS

Conventional OPC methods can be classified into two types, rule-based OPC [13, 14] and model-based OPC [3, 8]. Rule-based OPC is a simple approach by storing correction values for each representative pattern feature in advance [13, 14]. Because the pattern correction is based on rough and heuristic correction rules, rule-based OPC is a rapid approach but suffers from the limited correction accuracy especially in modern process nodes. In addition, rule-based OPC is hard to be applied to patterns with complex shapes and requires great effort to maintain the rules. On the other hand, model-based OPC calculates the displacements of fragmented pattern edges based on lithography simulation and iteratively refines mask patterns with edge-based mask optimization methods [3, 8]. The simulation iteration is continuously performed until the simulated image matches the desired one with acceptable edge placement errors (EPEs). EPEs are defined as the geometric displacements between the target polygon and the simulated pattern contour at measure points. Based on this process, model-based OPC achieves highly accurate correction but is extremely time-consuming due to computationally expensive simulations, especially for current complex and large-scale circuit layouts.

To tackle the run-time issue, some machine learning-based mask optimization methods have been proposed. A regression-based OPC approach has been proposed by using a hierarchical Bayes model (HBM) [12]. From their experimental results, it can be observed that the variance, standard deviation, mean, and median of EPE distribution achieve almost the same result as those of the 8^{th} iteration of model-based OPC, indicating that the number of optimization iterations can be reduced and runtime can be saved. On the other hand, Ma et al. and Xu et al. have proposed pixel-based SRAF generation methods respectively using non-parametric kernel regression and supervised data learning [11, 20]. A generative adversarial network (GAN) based mask optimization flow [7] have been developed by using pixel-based OPC masks as ground truth data as well. The generated quasi-optimal masks are expected to be a good initial input for the further inverse lithography technology (ILT) flow. However, in practical industrial application especially for sub-45nm process nodes, pixel-based mask optimization methods will result in prohibitely high mask complexity and are thus impracticle [10]. Observing the generated masks from pixel-based works, the complexity of OPCed mask and SRAF patterns are too high to be manufactured.

3 PRELIMINARIES

In this section, we first introduce the standard mask optimization flow. Further we define some terminologies applied in proposed framework, and some metrics in various aspects. Finally, we present the problem formulation of proposed framework.

3.1 Standard Mask Optimization Flow

A standard mask optimization flow [15] mainly includes two steps, SRAF insertion and model-based OPC, as shown in the left part of Figure 2. Given the original layout pattern as input, an SRAF insertion algorithm will firstly output the layout pattern with SRAF based on the analysis of the spacings among isolated or semi-isolated features. SRAF patterns are expected to enhance the process window without printing themselves on the wafer. After the completion of SRAF insertion, model-based OPC then calculates the displacements of fragmented pattern edges (not includes SRAF) based on lithography simulation and iteratively refines mask patterns. Eventually, model-based OPC outputs the final mask pattern consists of OPCed layout pattern and SRAF.

3.2 Optical Terminologies

In the lithography process, the photoresist is subjected to energy of intense light which comes from lithographic source and then passes through the mask, such that some chemical reactions are induced on photoresist. The regions subjected such chemical reactions are ultimately printed on the wafer through the following lithography steps. We introduce wafer image defined as follows:

Definition 1. *(**Wafer Image**) The area where the light intensity is greater than or equal to a specified threshold value on the wafer, which is shown as a grey region in Figure 3(a). The wafer image in Figure 3(a) is cut by a metrology plane, which is a user-defined plane cutting through a pattern that measures the values of various metrics at a specific site. The light intensity distribution on the cross-section of the metrology plane is shown in Figure 3(b) and the image is printed where the light intensity I exceeds the intensity threshold.*

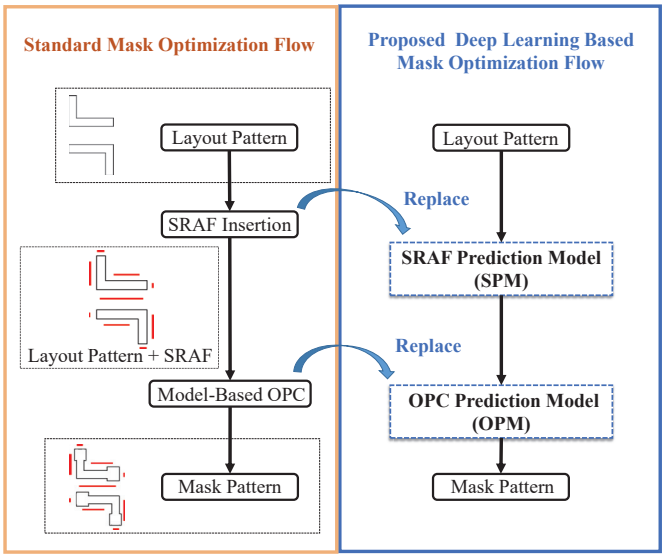

Figure 2: Standard mask optimization flow and proposed mask optimization flow.

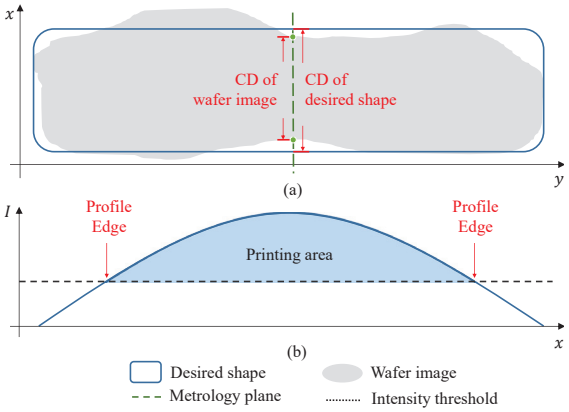

Figure 3: (a) Wafer image and CDs. (b) The light intensity distribution on the cross-section of the metrology plane in (a). The area whose intensity is greater than the threshold will be printed.

The general aim in mask optimization is to modify the given layout patterns so that the generated wafer image can be as close to the layout patterns as possible. Previous studies usually regard original layout patterns with sharp corners as the target images. However, this target standard is unrealistic and cannot be achieved in modern lithography process due to limited lithography resolution. Thus, we alternatively define our desired image shape as follows:

Definition 2. *(**Desired Shape**) The desired image shape of a polygon with rounded corners, whose radius is an empirically user-defined parameter. In this paper, we regard the original layout patterns with rounded corners (desired shapes) as our target images.*

In addition, due to sever process variations in advanced process nodes, the quality of wafer image can no longer be measured by a single metric.

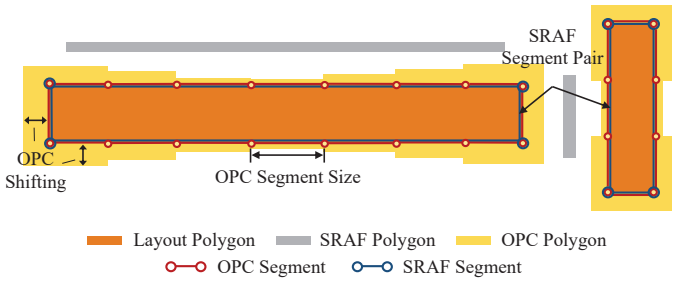

Figure 4: The introduction to OPC segments and SRAF segments.

The following metrics are adopted to evaluate mask optimization results in this work:

Definition 3. *(**Critical dimension error, CD error**) The error rate in percentage of the critical dimension (CD) of a wafer image compared to the CD of the corresponding desired shape, which can be computed as follows:*

$$CD\ Error = \left| \frac{CD_\alpha - CD_\beta}{CD_\beta} \right|$$

where CD_α denotes the CD of a wafer image, and CD_β denotes the CD of the desired shape.

Definition 4. *(**Depth of focus, DOF**) Depth of focus (DOF) means the total range of focus that can be tolerated; that is, the range of focus that keeps the resulting wafer image within a variety of specifications.*

Definition 5. *(**Process Window, PW**) Process Window is commonly used to assess the ability of tolerance to process variation of a design. We define process window as the maximal rectangular (or oval) area which is bounded by the tolerable change in DOF and the tolerable variation in exposure latitude to keep the wafer image within specified CD error.*

3.3 Problem Formulation

For performing edge-based SRAF insertion and OPC prediction, it is necessary to carry on whole processes from the viewpoint of single segment. Thus, we define **SRAF segment** and **OPC segment** as follows:

Definition 6. *(**SRAF segment**) Given a layout polygon, an SRAF segment is a single edge of a polygon. As shown in Figure 4, an SRAF segment represents the minimal unit to perform SRAF insertion. Two SRAF segments of the same direction (vertical or horizontal) which are closest to each other compose an SRAF segment pair. An SRAF can be inserted between each SRAF segment pair, and each SRAF segment pair records the position and dimension of the corresponding SRAF polygon if one is inserted.*

Definition 7. *(**OPC segment**) As shown in Figure 4, given a layout polygon, an OPC segment is a segment of a polygon edge with length equivalent to a user-defined OPC segment size. An OPC segment represents the minimal unit to perform edge-based OPC with the corresponding OPC shifting.*

To evaluate the performance of the deep learning-based SRAF insertion and edge-based OPC, we respectively define the following metrics:

Definition 8. *(**Root Mean Square Error** of deep learning-based SRAF insertion, $\mathrm{RMSE}_{\mathrm{SRAF}}$)*

$$\mathrm{RMSE}_{\mathrm{SRAF}} = \sqrt{\frac{\sum_{i=1}^{N}(y_i^* - y_i)^2}{N}}$$

where N is the total number of SRAF segments, y_i^* is the SRAF vector which records the corresponding the optimized SRAF's position and dimension information of the i^{th} SRAF segment, and y_i is the predicted SRAF vector. Note that the optimized SRAFs are generated by our random heuristic SRAF insertion algorithm according to the rule-based method [19], and the golden solution is set to be the one with best performance.

Definition 9. *(**Root Mean Square Error** of deep learning-based OPC, $\mathrm{RMSE}_{\mathrm{OPC}}$)*

$$\mathrm{RMSE}_{\mathrm{OPC}} = \sqrt{\frac{\sum_{j=1}^{n}(y_j^* - y_j)^2}{n}}$$

where n is the total number of OPC segments, y_j^* is the OPC shifting amount of the j^{th} single OPC segment derived from the model-based OPC and y_j is the predicted shifting amount.

With the above definitions, our problem formulations are defined as follows:

Problem 1. *(Deep learning-based SRAF insertion): Given an original layout, the objective of problem 1 is to generate the SRAF vectors with respect to the layout patterns. The goal of Deep learning-based SRAF insertion is to minimize* $\mathrm{RMSE}_{\mathrm{SRAF}}$.

Problem 2. *(Deep learning-based and edge-based OPC): Given an original mask with SRAF insertion, the objective of problem 2 is to generate a mask consisting OPCed patterns and SRAF vectors without any lithography simulation. The resulting mask is generated while minimizing* $\mathrm{RMSE}_{\mathrm{OPC}}$.

Our proposed framework is shown in the right part of Figure 2. *Problem 1* is to develop the SRAF Prediction Model (SPM) and *Problem 2* is to develop the OPC Prediction Model (OPM). The details will be discussed in Section 4.

4 DEEP LEARNING FRAMEWORK FOR SRAF INSERTION AND EDGE-BASED OPC

Because of the complex non-linear physical phenomena in lithographic process, using a linear prediction model such as linear regression is not sufficient [6]. Deep learning architecture has been proven to successfully model non-linear physical phenomena by composing of several successive layers of non-linear operations, such as neural nets with many hidden layers and various activation functions [16]. Also, for industry application in advanced process nodes, optimized masks with pixel-based mask optimization are not manufacturable due to their prohibitively high mask complexity.

4.1 Proposed Framework

To model the non-linear physical phenomena in mask optimization and to consider mask manufacturability, we propose a deep learning-based framework simultaneously considering SRAF insertion and edge-based OPC. The proposed mask optimization flow is shown in the right part of Figure 2. We replace the heuristic SRAF insertion and model-based OPC in the traditional mask optimization flow with the deep learning-based SRAF Prediction Model (SPM) and OPC Prediction Model (OPM) respectively. A well calibrated SPM is able to predict the exact SRAF pattern positions and dimensions for each SRAF segment pair. Using layout patterns as the input of SPM, SPM is able to generate the layout patterns with SRAFs. With the layout patterns with SRAFs as the input, OPM is trained to predict the displacement for each OPC segment to conduct an edge-based OPC and then outputs the final optimized mask.

4.2 Feature Extraction for Data

Because both SPM and OPM's prediction strategies are implicitly based on the proximity relationship between pattern and pattern, the way used to extract the proximity relationship plays such an important role on the performance of the models. Recently, concentric circle area sampling (CCAS) [11][12][20] has been proposed to extract the pattern proximity relationship and model light propagation. As shown in Figure 5, for OPM, each data point of an OPC segment represents a corresponding OPC displacement and several concentric sampling circles with different radii. Note that SRAF polygons are considered when executing feature extract

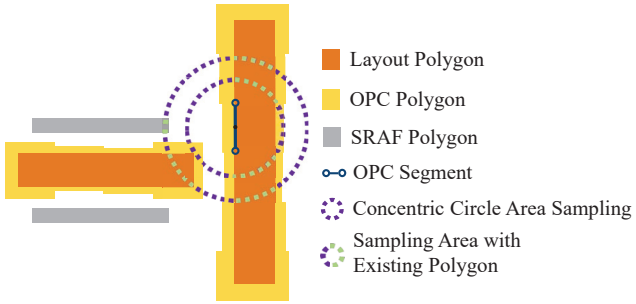

Figure 5: An OPC segment and the corresponding concentric sampling circle.

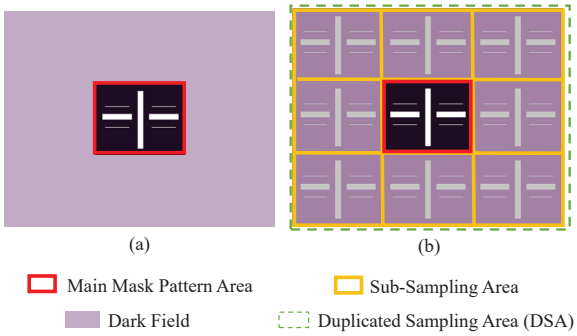

Figure 6: Main pattern area and duplicated sub-sampling area to avoid the sparseness in boarder sampling area.

for OPM. Similarly, for SPM, each data point of an SRAF segment pair represents the corresponding SRAF position and dimension information and several concentric sampling circles with different radii.

For calculating the radius of the j^{th} sampling circle R_j, we employ the equation: $R_j = r \times \alpha^j$ nm. Parameter r is the initial radius and α is a positive real number bigger then 1. Since the impact of the light diffraction is inversely proportional to the distance of proximity patterns, the sampling density should be much higher when we consider the closer neighboring area. Therefore, we apply the sampling density parameter α to control sampling density. Meanwhile, to avoid the extremely long run time, a mask pattern optimized in a commercial model-based OPC tool is usually a tiny clip from a full chip design. As shown in Figure 6(a), however, if we only execute the feature extraction in main mask pattern area, it may lead to the loss of proximity relationship from a global view because of the surrounding dark field. To tackle this issue, we propose the concept of duplicated sampling area (DSA), which duplicates the main pattern eight times and placing them around the main mask pattern area, as shown in Figure 6 (b). With the use of DSA, the better convergence is expected because the feature spareness caused by the dark field is mitigated and the proximity information also can be approximated more closely to a full chip design.

4.3 Training Data Preparation

Yin et al. [19] recently generalized the SRAF placement rules by analyzing the distance between each SRAF edge pair from commercial SRAF insertion results. With the simplified SRAF placement rules, they successfully improved process window with a straightforward rule-based SRAF method. The success of the rule-based SRAF place generalization [19] give us the opportunities to use some random configuration parameters to generate heuristic SRAF configurations for preparing SPM training data. We generalize the SRAF insertion principles for our heuristic SRAF insertion algorithm. The details of rules and parameters are listed in Table I. By considering the distance between an SRAF segment pair distance λ, we place random amount of SRAF polygons with random size between each SRAF segment. To make the SRAF insertion results more diverse, we generate N_r sets of SRAF insertion configurations for each benchmark, where for each SRAF segment pair with distance λ, at most three SRAFs are placed with randomly determined length L and distance D. In Table I, $S_0, S_1, S_2, S_3, S_4, S_5$ and W_0, W_1, W_2 are predetermined values derived from the generalized rules. With N_r sets of SRAF insertion results for each benchmark, we further choose the one with best PW SRAF as golden data. For OPM training data generation, we adopt the commercial model-based OPC tool PROLITH [22] to apply edge-based OPC. With the feature extraction techniques introduced in previous section, we introduce the process of our training data preparation for SPM and OPM. The details are explained in Algorithm 1.

Algorithm 1 Training Data Preparation

1: *Config* $C \leftarrow \{ C_1, C_2, \ldots, C_{N_r} \}$
▷ Random generated SRAF configuration parameter
2: *Layout Clip* $\mathcal{L} \leftarrow \{ \mathcal{L}_{1,1}, \mathcal{L}_{1,2}, \ldots, \mathcal{L}_{10,N_r} \}$
3: *Duplicated Sampling Area* $\mathcal{X} \leftarrow \{ \mathcal{X}_{1,1}, \mathcal{X}_{1,2}, \ldots, \mathcal{X}_{10,N_r} \}$
4: *Training Feature for SPM* $\mathcal{F}_{SPM} \leftarrow \phi$
5: *Training Feature for OPM* $\mathcal{F}_{OPM} \leftarrow \phi$
6: **procedure** SPM Training Data Generation
7: **for** *each* $\mathcal{L}_{t,u} \in \mathcal{L}, 1 \le t \le 10, 1 \le u \le N_r$ **do**
8: *SRAFEdges-Pairing*($\mathcal{L}_{t,u}, \mathcal{X}_{t,u}$);
9: $\mathcal{L}_{t,u} \leftarrow$*Heuristic-SRAF* ($\mathcal{L}_{t,u}, C_u$);
10: **end for**
11: **for** *each* $\mathcal{L}_{t,u} \in \mathcal{L}, 1 \le t \le 10, 1 \le u \le N_r$ **do**
▷ For each benchmark
12: $\mathcal{OPC}_{t,1}, \ldots, \mathcal{OPC}_{t,N_r} \leftarrow$Model-BasedOPC($\mathcal{L}_{t,1}, \ldots, \mathcal{L}_{t,N_r}$);
▷ Execute PROLITH model-based OPC
13: **end for**
14: **for** *each* $\mathcal{OPC}_{t,u} \in \mathcal{OPC}, 1 \le t \le 10, 1 \le u \le N_r$ **do**
15: $\mathcal{B}_t \leftarrow$ *TheBestSRAF*($\mathcal{OPC}_{t,1}, \ldots, \mathcal{OPC}_{t,N_r}$);
▷ Using PROLITH simulation to choose best PW SRAF mask for SPM golden data
16: $\mathcal{F}_{SRAF} \leftarrow$ *SRAF-FeatureExtraction*($\mathcal{B}_t, \mathcal{X}_{t,u}$);
17: $\mathcal{F}_{SPM} \leftarrow \mathcal{F}_{SPM} \cup \mathcal{F}_{SRAF}$
▷ Training data for SPM
18: **end for**
19: **end procedure**
20:
21: *TopGoodSRAF* $\mathcal{T} \leftarrow \phi$ ▷ $\mathcal{N}_{top}$ Layout clips with SRAF
22: **procedure** OPM Training Data Generation
23: **for** *each* $\mathcal{OPC}_{t,u} \in \mathcal{OPC}, 1 \le t \le 10, 1 \le u \le N_r$ **do**
▷ For each benchmark
24: $\mathcal{T} \leftarrow \mathcal{T} \cup$ *ChooseTopGoodSRAF*($\mathcal{OPC}_{t,1}, \ldots, \mathcal{OPC}_{t,N_r}, N_{top}$);
▷ Choose top $\mathcal{N}_{top}$ OPCed mask as OPM golden data
25: **end for**
26: **for** *each* $\mathcal{T}_t \in \mathcal{T}$ **do**
27: $\mathcal{F}_{opc} \leftarrow$ *OPC-FeatureExtraction*($\mathcal{T}_t, \mathcal{X}_{t,u}$);
28: $\mathcal{F}_{OPM} \leftarrow \mathcal{F}_{OPM} \cup \mathcal{F}_{opc}$ ▷ Training data for OPM
29: **end for**
30: **end procedure**

For the SPM training data generation, we first determine a user-defined parameter N_r to indicate the total amount of random generated SRAF placement configurations C (Line 1). C is the set of values L, W, D and S. Hence, for each of the 10 benchmarks of ICCAD 2013 CAD contest benchmark [21], we have N_r layout clips and duplicated sampling area (Line 2-3). In the process of SPM training data generation, we first generate all SRAF edge pairs by considering duplicated sampling area $\mathcal{X}$ and then use the N_r SRAF insertion configurations to generate the heuristic SRAF insertion results for each layout clip $\mathcal{L}_{t,u}$ (Line 7-10). Afterwards, for each benchmark, we execute PROLITH model-based OPC and then choose the best SRAF insertion result $\mathcal{B}_t$ with maximal process window as our golden SRAF and extract the corresponding SRAF training feature $\mathcal{F}_{SRAF}$ by considering duplicated sampling area $\mathcal{X}$ (Line 11-18). On the other hand, for OPM training data generation (from Line 21), to expand the number of training data for avoiding over-fitting, we choose the top N_{top} good SRAF insertion results for each OPCed masks (Line 23-25). The reason that we choose N_{top} samples for each benchmark is that there are too many prediction diversities of the SPM,

Table 1: Heuristic SRAF Insertion Principle Table

	SRAF Segment Pair Distance λ	SRAF Placement Result
Principle 1	$S_0 \leq \lambda \leq S_1$	An SRAF of width W_0 and length L is placed in the middle of the SRAF segment pair.
Principle 2	$S_1 \leq \lambda \leq S_2$	An SRAF of width W_1 and length L is placed in the middle of the SRAF segment pair.
Principle 3	$S_2 \leq \lambda \leq S_3$	Two SRAFs of width W_1 and length L are placed in distance D from SRAF segments in the SRAF segment pair respectively.
Principle 4	$S_3 \leq \lambda \leq S_4$	Place three SRAFs: 1. Two SRAFs of width W_1 and length L are placed in distance D from SRAF segments in the SRAF segment pair respectively. 2. An SRAF of width W_1 and length L is placed in the middle of the SRAF segment pair.
Principle 5	$S_4 \leq \lambda \leq S_5$	Place three SRAFs: 1. Two SRAFs of width W_2 and length L are placed in distance D from SRAF segments in the SRAF segment pair respectively. 2. An SRAF of width W_2 and length L is placed in the middle of the SRAF segment pair.

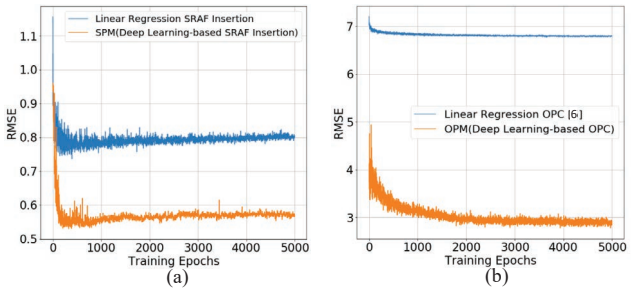

Figure 7: Validation RMSE curve (a) SRAF insertion model (b) OPC model

so there is a strong demand for increasing diversities of OPM training data to make the predicted results of OPM more robust. Eventually, we consider the duplicated sampling area $\mathcal{X}$ to obtain $\mathcal{F}_{opc}$ as the training features of OPM (Line 26-29).

5 EXPERIMENTAL RESULTS

We implemented our deep learning framework for SRAF insertion and edge-based OPC in python with TensorFlow library [1]. Our framework is trained and tested on Nvidia GTX1080Ti graphic card and Intel i7-8700K 3.70GHz processor machine with 32GB memory. We performed CCAS feature extraction in C++ programming language. The testbenches are ten industrial 32nm node M1 designs which are provided by ICCAD 2013 CAD contest [21]. CAD contest only provides lithography simulator but no golden masks, which means it is difficult for us to produce our training data without authoritative model-based OPC algorithm. Also, the provided lithography simulator is pixel-based and thus not applicable to our framework. Thus, we use the commercial lithography tool PROLITH [22] instead of the provided lithography simulator to generate the model-based OPC training data set and simulate the lithography process . The evaluation flow is automatized by calling the application program interface (API) of PROLITH by MATLAB on an Intel i5-4570 3.50GHz processor machine with 8GB memory. For SPM and OPM, the neural networks are trained by following network configuration: five hidden layers with ReLU activation functions. 50% dropout layers and $L2$ regularizers are applied on the hidden layers to prevent overfitting. Training optimizer is AdaGrad and the total number of training epochs is 5000.

In the first experiment, we examine the lithography performance of our gloden SRAF plus model based- OPC by checking the PW overlapping area and depth of focus (DOF). We set CD error within 10% as our specifications, which is acceptable in industrial application [4]. Table 2 summarizes the results, where "PW O.L." denotes the PW overlapping area and "RT" gives the runtime in second. It can be observed that PW overlapping area and DOF are both increasing by about 30% after adding SRAFs, which is a significant improvement with acceptable trade-off in running time.

After checking the effectiveness of our data, we employ Algorithm 1 and the output masks of "Gloden SRAF+Model-Based OPC" in Table 2 to generate training data and testing data for SPM and OPM. Benchmarks 1-7 are treated as training data and benchmarks 8-10 as testing data. We set $N_r = 100$ and $N_{top} = 30$ in Algorithm1. For the CCAS feature extraction parameter selection, we test the parameter initial radius r, sampling density parameter α and the total number of sampling circle $Circle$ # within a range and choose the best parameter configuration which can achieve the best loss on testing data set. The details of the optimal parameter configuration are listed in Table 3. After that, we compare the root mean square error (RMSE) curve of the proposed framework with that derived from linear regression. The curves are shown in Figure 7, where the X-axis is the training epochs and the Y-axis is the RMSE of the testing set. From Figure 7, we can observe that, for testing data, both SPM and OPM achieve the better convergence with a lower RMSE loss then linear regression-based SRAF insertion and linear regression-based OPC [6]. Note that the parameter configuration of CCAS feature extraction for linear regression is also optimized.

Finally, we check the optimized masks of SPM plus OPC and the lithography simulation results are listed in Table 4. From the third row in Table 4, the optimized masks increase 16% PW overlapping area and 6% DOF with 92% runtime improvement compared "Model-Based OPC". Compared to "Gloden+Model-Based OPC", whose results are regarded as golden solutions, our framework also achieves similar performance with significant runtime improvement. The average CD error derived from our framework is averagely 5%, which is under 10% (the industrial standard [4]). The optimized masks are visualized in Figure 8, where the three testing cases are respectively showed. Figure 8(a) gives the layout pattern, which is the original target layouts. The masks with SRAF inserted by our SPM are shown in Figure 8(b), and Figure 8(c) shows the masks where the model-based OPC is applied after performing SPM. Figure 8(d) shows the optimized masks after SPM and OPM, whose generated high quality wafer images are also given in Figure 8(e).

6 CONCLUSION

In this paper, we proposed a novel deep learning framework for comprehensive mask optimization with SRAF insertion and edge-based OPC. We simultaneously enhance the printability of a design and the tolerance to process variations. We further consider the manufacturability of a mask by performing edge-based OPC that make it more practical and realistic for industrial application. The deep learning framework greatly shows its advantage so that the SRAF insertion and edge-based OPC can be completed without any simulation, achieving a more efficient RET process. Experimental results show that the process window of an optimized mask is enhanced after our deep learning-based SRAF insertion and edge-based OPC are accomplished. At the same time, the runtime is remarkably reduced compared with the traditional mask optimization flow.

Table 2: Lithography Performance of Training Data Set

	Benchmarks		Training Data 1	2	3	4	5	6	7	Avg.	Ratio
1	MB OPC	**PW O.L.**	17	23.1	7.74	20	21	15.6	27	18.78	1.00
		DOF (um)	0.09	0.12	0.04	0.11	0.11	0.08	0.14	0.1	1.00
		RT(s)	1.61	1.09	2.11	1.19	2.14	1.91	1.63	1.67	1.00
2	Golden SRAF + MB OPC	**PW O.L.**	23.16	28.3	20.23	25.85	24.88	17.53	34.16	24.87	1.32
		DOF (um)	0.12	0.14	0.11	0.13	0.13	0.09	0.18	0.13	1.3
		RT (s)	3.67	1.18	2.09	0.86	2.04	1.83	2.2	1.98	1.18

Table 3: Optimum parameter configuration for CCAS feature extraction

	CCAS Parameter α	r	*Circle#*
SPM	1.2	25	9
OPM	1.25	20	13

Table 4: Lithography Performance of Testing Data Set

	Benchmarks		Testing Data 8	9	10	Avg.	Ratio
1	MB OPC	**PW O.L.**	20.46	22.71	42.8	28.66	1.00
		DOF (um)	0.11	0.12	0.21	0.15	1.00
		RT(s)	1.09	1.85	0.68	1.2	1.00
2	Golden SRAF + MB OPC	**PW O.L.**	27.82	29.55	52.08	36.49	1.27
		DOF (um)	0.14	0.15	0.26	0.18	1.2
		RT (s)	1.07	2.06	0.77	1.3	1.09
3	SPM SRAF + OPM OPC	**PW O.L.**	25.75	26.74	46.96	33.15	**1.16**
		DOF (um)	0.12	0.13	0.22	0.16	**1.06**
		RT (s)	0.07	0.14	0.05	0.09	**0.08**
		CDE (%)	4.54	8.55	1.92	5	

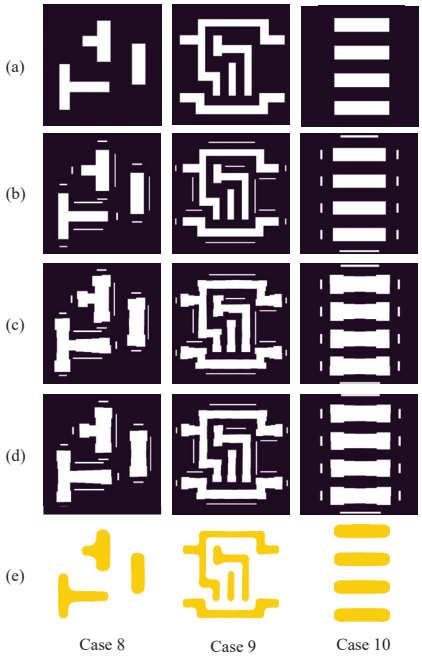

Figure 8: Resulting masks of (a) Layout pattern, (b) SRAF inserted by SPM, (c) SPM SRAF with model-based OPC, and (d) SPM SRAF with OPM OPC. (e) shows the resulting wafer image of our SPM SRAF with OPM OPC. Columns of the figure respectively represent the three testing cases.

ACKNOWLEDGEMENT

The authors would like to thank KLA-Tencor for providing PROLITH software to complete the numerical work.

REFERENCES

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean et al., "TensorFlow: A system for large-scale machine learning," in Proc. OSDI, pp. 265âĂŞ283, 2016.
[2] Kuan-Jung Chen, Yu-Kai Chuang, Bo-Yi Yu and Shao-Yun Fang, "Minimizing cluster number with clip shifting in hotspot pattern classification," 2017 Design Automation Conference (DAC), pp. 1-6, 2017.
[3] N. Cobb, "Fast optical and process proximity correction algorithm for integrated circuit manufacturing," Ph.D. dissertation, University of California at Berkeley, 1998.
[4] F. Lie, C. Y. Huang, C. S. Wu, K. T. Chen and H. F. Kuo, "Demonstration of ACO-Based Freeform Source for ArF Laser Immersion Lithography System," in IEEE Access, vol. 5, pp. 6421-6428, 2017.
[5] A. H. Gabor, J. A. Bruce, W. Chu, R. A. Ferguson, C. A. Fonseca, R. L. Gordon, K. R. Jantzen, M. Khare, M. A. Lavin, W. Lee, L. W. Liebmann, K. P. Muller, J. H. Rankin, P. Varekamp, and F. X. Zach, "Subresolution assist feature implementation for high-performance logic gate-level lithography," in *Proceedings of SPIE*, vol. 4691, pp. 418–425, 2002.
[6] A. Gu and A. Zakhor, "Optical proximity correction with linear regression," *IEEE Transactions on Semiconductor Manufacturing*, vol. 21, no. 2, pp. 263–271, 2008.
[7] Haoyu Yang, Shuhe Li, Yuzhe Ma, Bei Yu and Evangeline F. Y. Young, "GAN-OPC: Mask Optimization with Lithography-guided Generative Adversarial Nets," 2018 Design Automation Conference (DAC), Article No. 131, 2018.
[8] S.-K. Kim, "Model-based OPC for resist reflow process," *Digest of Papers Microprocesses and Nanotechnology*, pp. 18–19, 2005.
[9] R. Luo, "Optical proximity correction using a multilayer perceptron neural network," *Journal of Optics*, vol. 15, no. 7, 2013.
[10] X. Ma and Y. Li, "Resolution enhancement optimization methods in optical lithography with improved manufacturability," *Journal of Micro/Nanolithography, MEMS, and MOEMS*, vol. 10, no. 2, 2011.
[11] X. Ma, B. Wu, Z. Song, S. Jiang, and Y, Li, "Fast pixel-based optical proximity correction based on nonparametric kernel regression," *Journal of Micro/Nanolithography, MEMS, and MOEMS*, vol. 13, no. 4, 2014.
[12] T. Matsunawa, B. Yu, D. Z. Pan, "Optical proximity correction with hierarchical bayes model," in *Proceedings of SPIE*, 2013.
[13] O. W. Otto, J. G. Garofalo, K. K. Low, C.-M. Yuan, R. C. Henderson, C. Pierrat, R. L. Kostelak, S. Vaidya, P. K. Vasudev, "Automated optical proximity correction: a rules-based approach," *in Proceedings of SPIE*, 1994.
[14] J.-S. Park, C.-H. Park, S.-U. Rhie, Y.-H. Kim, M.-H. Yoo, J.-T Kong, H.-W. Kim, and S.-I. Yoo, "An efficient rule-based OPC approach using a DRC tool for 0.18 Îijm ASIC," in *Proceedings IEEE International Symposium on Quality Electronic Design*, 2000.
[15] Y. Ping et al., "Process window enhancement using advanced RET techniques for 20nm contact layer," in *Proceedings of SPIE*, Art. no. 90521N, 2014.
[16] J. Schmidhuber. "Deep Learning in Neural Networks: An Overview. Neural Networks," Volume 61, Pages 85-117, January 2015.
[17] Y.-H. Su, Y.-C. Huang, L.-C. Tsai, Y.-W. Chang, and S. Banerjee, "Fast lithographic mask optimization considering process variation," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 8, pp. 1345–1357, 2016.
[18] A. K. Wong, "Resolution enhancement techniques in optical lithography," in *Proceedings of SPIE*, 2001.
[19] L. Yin, C. Wang, and L. Li, "Process window tripling by optimized SRAF placement rules: AP/DFM: Advanced patterning/design for manufacturability," in *Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC)*, 2016.
[20] X. Xu, Y, Lin, M. Li, T. Matsunawa, S. Nojima, C. Kodama, T. Kotani, and D. Z. Pan, "Sub-resolution assist feature generation with supervised data learning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2017.
[21] S. Banerjee, Z. Li, and S. R. Nassif, "ICCAD-2013 CAD contest in mask optimization and benchmark suite," in *Proceedings of International Conference on Computer-Aided Design*, 2013.
[22] KLA-Tencor PROLITH user guide. https://www.kla-tencor.com/