From IC Layout to Die Photograph: A CNN-Based Data-Driven Approach

Hao-Chiang Shao, *Member, IEEE*, Chao-Yi Peng, Jun-Rei Wu, Chia-Wen Lin, *Fellow, IEEE*, Shao-Yun Fang, *Member, IEEE*, Pin-Yian Tsai, and Yan-Hsiu Liu

***Abstract*—We propose a deep learning-based data-driven framework consisting of two convolutional neural networks: 1) LithoNet that predicts the shape deformations on a circuit due to IC fabrication and 2) OPCNet that suggests IC layout corrections to compensate for such shape deformations. By learning the shape correspondences between pairs of layout design patterns and their scanning electron microscope (SEM) images of the product wafer thereof, given an IC layout pattern, LithoNet can mimic the fabrication process to predict its fabricated circuit shape. Furthermore, LithoNet can take the wafer fabrication parameters as a latent vector to model the parametric product variations that can be inspected on SEM images. Besides, traditional optical proximity correction (OPC) methods used to suggest a correction on a lithographic photomask is computationally expensive. Our proposed OPCNet mimics the OPC procedure and efficiently generates a corrected photomask by collaborating with LithoNet to examine if the shape of a fabricated circuit optimally matches its original layout design. As a result, the proposed LithoNet–OPCNet framework can not only predict the shape of a fabricated IC from its layout pattern but also suggests a layout correction according to the consistency between the predicted shape and the given layout. Experimental results with several benchmark layout patterns demonstrate the effectiveness of the proposed method.**

Manuscript received February 10, 2020; revised May 25, 2020; accepted July 16, 2020. Date of publication August 10, 2020; date of current version April 21, 2021. This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 108-2634-F-007-009; and in part by United Microelectronics Corporation. This article was recommended by Associate Editor L. Behjat. *(Corresponding author: Chia-Wen Lin.)*

Hao-Chiang Shao is with the Department of Statistics and Information Science, Fu Jen Catholic University, New Taipei City 24205, Taiwan (e-mail: shao.haochiang@gmail.com).

Chao-Yi Peng was with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan. He is now with the Image Signal Processing Team, Altek Corporation, Hsinchu 300, Taiwan (e-mail: sky135410@yahoo.com.tw).

Jun-Rei Wu was with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan. He is now with the Advanced Creativity Team, HTC VIVE, New Taipei City, Taiwan (e-mail: heathentw@gmail.com).

Chia-Wen Lin is with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan, and also with the Institute of Communications Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: cwlin@ee.nthu.edu.tw).

Shao-Yun Fang is with the Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei 10607, Taiwan (e-mail: syfang@mail.ntust.edu.tw).

Pin-Yian Tsai is with the Product Engineering Division, United Microelectronics Corporation, Hsinchu 300, Taiwan (e-mail: pin_yian_tsai@umc.com).

Yan-Hsiu Liu is with the Smart Manufacturing Division, United Microelectronics Corporation, Hsinchu 300, Taiwan (e-mail: cecil_liu@umc.com).

Digital Object Identifier 10.1109/TCAD.2020.3015469

***Index Terms*—Convolutional neural networks (CNNs), design for manufacturability, lithography simulation, optical proximity correction (OPC), virtual metrology (VM).**

I. Introduction

AFTER IC circuit design and layout, it typically takes two to three months to fabricate a 12-inch IC wafer, involving a multistep sequence of photolithographic and chemical processing steps. Among these steps, a lithography process is used to transfer an IC layout pattern from a photomask to a photosensitive chemical photoresist on the substrate, followed by an etch process that chemically removes parts of a polysilicon or metal layer, uncovered by the etching mask, from the wafer surface. Because it is hard to control the exposure conditions and the chemical reactions involved in all fabrication steps, the two processes together lead to nonlinear shape distortion of a designed IC pattern, which is usually too complicated to model. This fact urges the need for *mask optimization*, a procedure that computes an optimized photomask to make the shape of the fabricated IC wafer optimally consistent with its source layout design.

The inevitable shape deformations on a fabricated IC due to the imperfect lithography and etch processes often cause IC defects (e.g., thin wires or broken wires) if an IC circuit layout is not appropriately designed, especially on the first few metal layers. Nevertheless, in most cases, we still cannot identify such IC defects due to inappropriate IC circuit layout until capturing and analyzing the scanning electron microscope (SEM) images of metal layers after the wafer fabrication process, making the circuit verification very costly and time consuming. It is therefore desirable to develop presimulation tools, including: 1) a lithography simulation method for predicting the shapes of fabricated metal lines based on a given IC layout along with IC fabrication parameters and 2) a mask optimization strategy for predicting the best mask to compensate for the shape distortions caused by the lithography and etch processes.

As for lithography simulation, there are two categories of conventional approaches: 1) physics-level rigorous simulation and 2) compact model-based simulation [1], [2]. Rigorous simulation methods simulate physical effects of materials to accurately predict a fabricated circuit and thus are very time consuming [3], [4]. On the contrary, a compact model-based simulation method follows loosely physical phenomena to obtain a faster computational speed by exploiting complicated, parameter-dependent, nonlinear functions. Different from traditional methods, we aim at developing a convolutional neural network (CNN)-based approach, which learns the

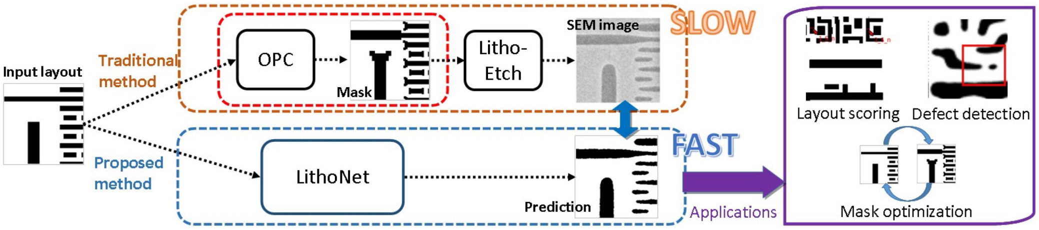


Fig. 1. Relationship among OPC simulation, circuit verification on an SEM image, and our method. The OPC step, highlighted by the red dashed lines, suggests modifications of a layout mask so that the fabricated IC could have nearly the same shape as the original layout pattern. The proposed LithoNet and its applications are highlighted by purple contours.

parametric model of physical and chemical phenomena of a fabrication process directly from a training dataset containing pairs of IC layouts and their corresponding SEM images. Based on the learned CNN model, we can predict a fabricated circuit shape more accurately and efficiently than traditional methods.

Moreover, fab-engineers usually optimize a mask pattern by iteratively modifying a layout design based on its lithography simulations. However, rule-based lithography simulations resort to linear combinations of optical computations derived from several similar yet not identical historical fab-models. The simulation reliability largely relies on a rich amount of costly historical fabrication data because ground-truth fab-models need to be gathered by fabricating a layout pattern with all process variations exhaustively. Nevertheless, fab-plants do not typically build models with exhaustive data but, instead, select nominal plus some relatively small number of specific process-window conditions over a limited number of test structures and then build models based on that. This fact may make current standard unreliable for new layout design patterns.

The relationship among the IC fabrication process, lithography simulator, and mask optimizer is depicted in Fig. 1, where the optical proximity correction (OPC) block is a standard approach to photomask correction for compensating for the shape distortions due to diffraction or process effects as well as guaranteeing the printability of a layout pattern, especially at the corners of the process window [5], [6]. As shown in the red dashed rectangles in Fig. 1, the mask used in the fabrication process is a modified version of a source layout design, aiming to compensate for possible "shrinkages" in line shapes due to the fabrication to mitigate the deviation of a fabricated IC circuitry from its layout design. However, traditional OPC methods have two primary drawbacks. First, they run simulations based on those rules and patterns already known; thus, an OPC correction may be unreliable if an unseen layout design is given. Second, not only is the OPC correction computationally expensive but also the OPC contour simulation is a time-consuming trial-and-error routine that is iterated until no irregularity can be found in the OPC estimation result. Both the OPC correction and OPC contour simulation are computationally expensive. Take the *ICWB* software (IC WorkBench) developed by Synopsys [7] for example. ICWB takes, on average, about 34 s to run a contour simulation on a 4 × 1.7 μm^2 layout patch with an Intel Xeon E5-2670 CPU and 128-GB RAM. It will cost around 4 days to run one OPC contour simulation on a 400 × 170 μm^2 layout design, and such computational cost makes a complete OPC contour simulation procedure impractical. It is therefore highly desirable to develop an efficient photomask optimization scheme.

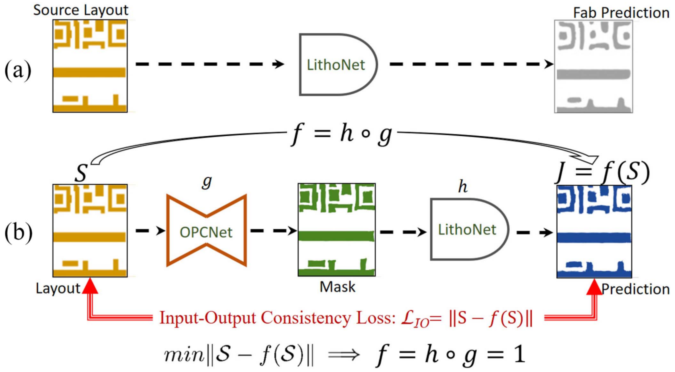

Source Layout
(a)
LithoNet
Fab Prediction
$f = h \circ g$
S
g
h
$J = f(S)$
(b)
OPCNet
LithoNet
Layout
Mask
Prediction
Input-Output Consistency Loss: $\mathcal{L}_{IO}$= $\|S - f(S)\|$
$min\|\mathcal{S} - f(\mathcal{S})\| \implies f = h \circ g = 1$

Fig. 2. Two scenarios utilizing the proposed LithoNet and OPCNet: (a) standalone LithoNet and (b) cascaded LithoNet–OPCNet network.

Recent progress on image-to-image translation techniques makes them suitable to tackle the lithography simulation (i.e., Layout-to-SEM) and photomask optimization (i.e., SEM-to-Layout) problems mentioned above. However, these two issues are more complicated than general image-to-image translation problems. Take Layout-to-SEM prediction for example. First, the domain of IC layout images and that of SEM images are heterogeneous. An IC layout is a purely man-made blueprint with only lines and rectangles on it and, hence, it is noise-free and artifact-free. On the contrary, an SEM image is formed from the intensity of detected signal from raster-scanning the IC surface with a focused electron beam. Besides the continuous shape distortions introduced by the lithography and etching processes, the SEM imaging process itself also suffers from several kinds of interference (e.g., scan-line noise and shading). This fact leads SEM images to a significantly different domain from the layout-image domain. Hence, this issue is essentially a cross-domain image matching and translation problem. Second, in order to predict the corresponding SEM image from an IC layout, our solution must be capable of finding the shape correspondence between these two domains of images. This fact raises an unsupervised cross-domain image matching issue, which usually has not been concerned in general image-to-image translation techniques. Thus, it requires a more sophisticated solution, as the concerns stated in [8] and [9]. Third, for the mask optimization problem, it is very costly to collect a comprehensive set of reference OPC-corrected photomasks, making the training of a photomask optimization network infeasible.

To address the above problems, as shown in Fig. 2, we propose a fully data-driven framework involving two CNNs, LithoNet and OPCNet, functionally complementary to each other. In short, LithoNet is a cross-domain simulator of the lithography and etch processes in IC fabrication, and OPCNet is a self-supervised mask optimization CNN using the prediction results of LithoNet as supervision for the purpose of OPC. The proposed LithoNet–OPCNet network serves two purposes, each requiring a specific training dataset. First, when LithoNet is used stand-alone as shown in Fig. 2(a), it aims at image-to-image contour prediction. Because we focus on the Layout-to-SEM (or Mask-to-SEM) contour prediction problem, we train LithoNet on (layout, SEM) data pairs. Then, during the inference stage, given a layout design, LithoNet predicts: 1) a deformation map and 2) an SEM prediction. Both the deformation map and SEM prediction can be used for layout risk assessment. Note that LithoNet is an image-to-image contour predictor and thus can be trained on different kinds of paired images for different purposes. For example, if we need to build a model for mask-to-SEM prediction, we have to train LithoNet on (mask, SEM) data pairs.

Second, as shown in Fig. 2(b), when OPCNet and LithoNet are cascaded, the LithoNet–OPCNet network forms a system for mask optimization aiming at minimizing the discrepancy between a source layout and its SEM contour predicted by LithoNet. The design concept is to construct a two-stage system, where the first stage performs layout-to-X prediction by OPCNet, where X denotes the OPC-corrected mask, and the second stage performs X-to-SEM prediction by LithoNet. Then, by enforcing the SEM prediction to be shape-consistent with the target layout (i.e., the whole OPCNet-LithoNet network behaves as an identity transform), OPCNet and LithoNet act as if they were inverse functions of each other mathematically. As a result, the LithoNet–OPCNet network can be used to find an OPC-optimized mask X.

This article has four primary contributions.

1) To the best of our knowledge, we are the first to formulate the Layout-to-SEM deformation prediction problem as a cross-domain image correspondence problem, and we propose a two-step CNN-based framework to address it.
2) Our LithoNet–OPCNet system is computationally much more efficient than the typical optical-based contour simulation schemes, while achieving comparable prediction accuracy. Since our method is fully data-driven, it could enable IC fabrication plants to run a full, large-scale screening on new IC layout designs. Note that an OPC model is typically built for a particular process condition and operates according to interpolation. Hence, if the process condition changes for a given process node, whether the input layout is completely new or not, the same OPC model may not provide a reliable prediction.
3) The proposed LithoNet is parameterized with fabrication settings. Hence, it can also predict results under different fabrication conditions so as to assist fabrication plants to find the best suitable working intervals of parameters and thus be beneficial for yield-rate improvement.
4) The proposed OPCNet overcomes the difficulty in lack of ground-truth mask patterns. With the aid of a novel training objective function called *I/O-consistency* loss, the proposed OPCNet can well simulate the mask optimization process in collaboration with LithoNet.

The remainder of this article is organized as follows. We review related literature in Section II. The proposed LithoNet and OPCNet are detailed in Sections III and IV, respectively. Section V demonstrates and discusses our experimental results. Finally, we draw our conclusion in Section VI.

II. Related Work

A. Virtual Metrology

In IC fabrication, virtual metrology (VM) refers to the methods for predicting wafer properties based on fabrication parameters and sensor data from equipment without performing physical measurements on the product wafer produced by a whole, costly fabrication process [10]. Since VM techniques can significantly reduce the cost of IC fabrication, various kinds of VM methods have been proposed for fabrication quality assessment. For example, Susto *et al.* exploited the knowledge collected in the process steps to improve the accuracy of VM prediction via a multistep strategy [11]. Besides, the demand of VM methods has also triggered the development of theoretical techniques. The method in [12], for instance, models OPC mask correction as an inverse problem of optical microlithography. Optical lithography is a process used for transferring binary circuit patterns onto silicon wafers, and related discussions about lithography techniques can be found in [13]. Recently, people have been attempting to integrate machine learning methods with IC implementation and VM [1], [2], [14]–[16]. Specifically, Yang *et al.* [15] proposed a generative adversarial network (GAN) [17]-based inverse method to estimate the optimal mask used in the fabrication process from an OPC simulation result. However, the design in [15] aims only at the OPC-to-Layout problem, which operates in an opposite direction of our Layout-to-SEM prediction. Therefore, to the best of our knowledge, there is no existing technique focusing simultaneously on both Layout-to-SEM (lithography simulation) and SEM-to-Layout (mask optimization) image translation problems. We deem that a hybrid method of image-to-image translation or feature mapping techniques could compose a straightforward solution to these two prediction problems.

B. Lithography Simulation

Recently, there have been a few machine learning-based lithography simulation methods. For instance, Watanabe *et al.* [1] proposed a fast and accurate lithography simulation by determining an appropriate model function via CNN, and Ye *et al.* [2] developed a GAN-based end-to-end lithography modeling framework, named LithoGAN, to map directly the input mask pattern to the output resist pattern. Specifically, LithoGAN models the shape of the resist pattern based on a conditional GAN (cGAN) model and predicts the center location of the resist pattern via a CNN model. LithoGAN has a dual learning framework and, similarly, our LithoNet also adopts a dual learning framework.

As will be detailed in Section III, we formulate the Layout-to-SEM prediction as a cross-domain image-to-image translation problem in the LithoNet design. Recent image-to-image translation methods can be divided into two groups. One requires training image pairs, e.g., [18] and [19], and the other supports training on unpaired data, e.g., [20]. The method in [20], based on GANs [17] and VAEs [21], was designed for unsupervised image-to-image translation tasks, which could be considered as a conditional image generation

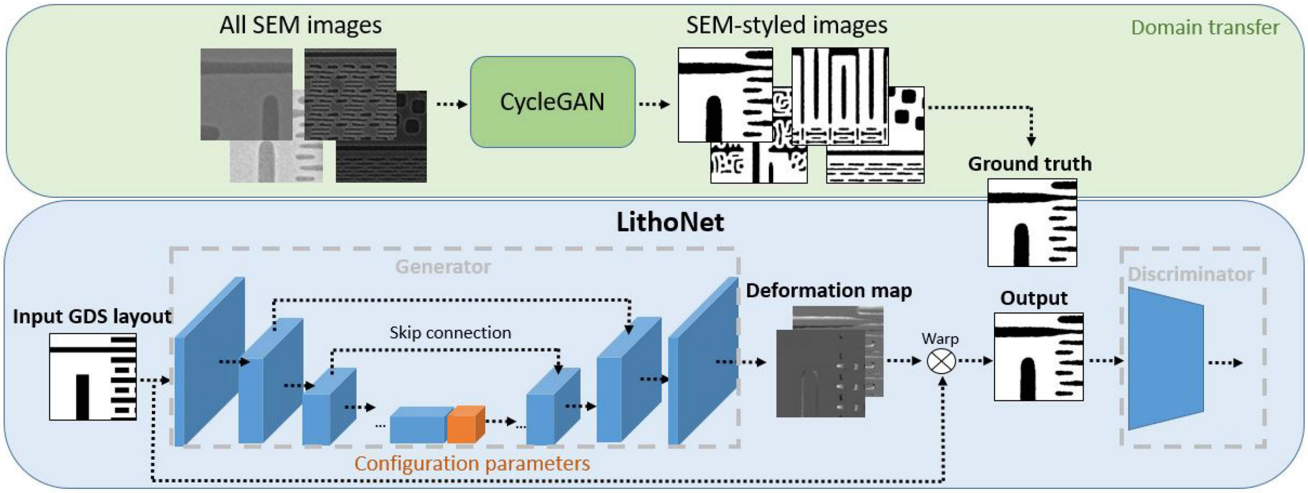


Fig. 3. Block diagram of the proposed two-step framework for cross-domain image-to-image translation. The upper step adopts CycleGAN to transfer the training SEM images to obtain ground-truth labels. LithoNet then estimates the deformation maps between input layout patterns and their corresponding labels.

model. Besides, Pix2pix [18] consists of a Unet-like generator and a PatchGAN discriminator. Pix2pix uses the PatchGAN discriminator to model high-frequencies by classifying if each patch in an image is real or fake. Therefore, it can be adopted in various applications, such as translating a cartoon map to a satellite image and translating a sketch to a natural image and has become a benchmark in this field. Pix2pix was further enhanced in [19] by taking advantage of a course-to-fine generator, a multiscale discriminator, and an adversarial learning objective function so as to generate high-resolution photo-realistic images.

However, the above methods cannot address the shape correspondence and the deformation field between two different domains of images, and neither do other representative image-to-image translation methods, such as CycleGAN [22], DualGAN [23], and [20], [24], [25]. Because characterizing the deviations of metal lines on a product IC based on the source layout is a critical point in the IC industry, traditional image-to-image translation methods, which lack a mechanism for precisely estimating a deformation field or the shape correspondence between the layout and SEM images, are not applicable to Layout-to-SEM image translation. To serve the above purpose, the proposed LithoNet model performs cross-domain image-to-image translation via learning the shape correspondence between paired training images so as to output a predicted deformation map for further VM applications.

C. Mask Optimization

There also exist machine learning-based mask optimization approaches. Notably, GAN-OPC proposed in [15] takes source layout patterns and their reference OPC photomasks as training inputs and accordingly, for an input layout design, predicts a corrected photomask that minimizes the deviation on the (simulated) fabricated circuit shape from its original design. In order to facilitate the training process and guarantee convergence, GAN-OPC involves a pretrain procedure that trains jointly the neural network and the inverse lithography technique (ILT) [26]. After GAN-OPC converges, the obtained quasi-optimal photomask is further used as a reasonable initial estimate for further ILT operation. In contrast, Yu *et al.* [16] proposed a DNN framework to simultaneously perform sub-resolution assist feature (SRAF) [27] and edge-based OPC. However, the two methods require a collection of photomask images, such as those suggested by OPC or historical data gathered during the actual fabrication process, as the ground-truth dataset for training. Because it is expensive and time consuming to collect qualified mask images, the cardinality of the training dataset forms a performance bottleneck of these methods. To eliminate such a bottleneck, we propose the OPCNet model for mask optimization, powered by LithoNet. Because OPCNet and LithoNet are the inverse function to each other, OPCNet can be trained directly on the SEM-styled images predicted by LithoNet without the need for using expensive photomask images, as will be elaborated later.

III. LithoNet: A CNN-Based Lithography Simulator

As shown in Fig. 3, LithoNet consists of a CycleGAN-based [22] domain transfer network and a deformation prediction network. LithoNet is designed to learn how an IC fabrication process deforms the shape contours of a layout pattern. It can simulate the fabrication process to predict the shape deformation for further VM applications based on: 1) a given layout and 2) a set of fabrication parameters. One major difficulty in learning the shape deformation model between a layout pattern and its corresponding SEM image of fabricated circuitry lies in the fact that they are from heterogeneous domains. Specifically, an SEM image is a high-resolution, gray-scaled image with deep depth of field (DOF), whereas a layout is no more than a man-made binary pattern with only rectangular regional objects on it. As a result, the goal of LithoNet is to predict the contour shapes by learning the pixelwise shape correspondence between every paired layout and SEM images. Nevertheless, due to the poor contrast and scanning pattern noise in SEM images, it is usually difficult to capture edge contours correctly from SEM images, on which a 1-pixel-drift corresponds to a nanometer-scale displacement on real IC products. Therefore, transferring the domain of SEM images to another intermediate domain without the above-mentioned contrast and noise problems would be beneficial.

To this end, we propose a two-step framework. In the first step, we use CycleGAN [22] to transfer a gray-scale SEM image to an intermediate domain, where images have SEM-styled shape contours and layout-styled clear background. Then, in the second step, given a source layout along with fabrication parameters, LithoNet predicts the shape deformation introduced by the fabrication process. In sum, Step I learns to bridge the gap between the SEM image and its binary layout so that Step II can learn the shape correspondence between the SEM image and its original layout. In the following sections, we will introduce our design in detail.

A. Step I: Image Domain Transfer

Because the SEM and layout images are of heterogeneous domains, we adopt an image domain transfer technique to *align* their domains. By removing the interference introduced by the SEM imaging process, e.g., bias in brightness/contrast and scan-line noise, via CycleGAN [22], the processed SEM image is translated to the domain of the layout. That is, the processed SEM image retains its curvilinear shape boundaries yet is binarized as if it were a layout.

To this end, we train CycleGAN using: 1) a set of product-ICs' SEM images and 2) their associated segmentation masks. The second set of images can be derived by applying either manual labeling, advanced thresholding techniques [28], [29], interactive segmentation [30], [31], or pseudo-background subtraction [32] on the source SEM images. Note that in order to guarantee the performance of domain transfer, segmentation masks with incorrect segmentation results are discarded under user-supervision. Finally, we utilize the well-trained CycleGAN to transfer source SEM images into the layout style, and these processed SEM images are further taken as reference ground truths to train LithoNet in Step II.

Employing CycleGAN for domain transfer has two advantages. First, CycleGAN is an unpaired image-to-image translation method and, hence, it can learn the majority decision of multiple image segmentation algorithms, including the analysis software provided by the SEM vendor, for SEM images based on a collection of segmentation results of different methods. Second, utilizing a "U-net Generator" to translate images, CycleGAN is essentially a U-net-based segmentation method [33] supervised by its built-in "Discriminator" through an adversarial loss, thereby suggesting a more reliable segmentation result than U-net, a state-of-the-art segmentation benchmark. Additionally, we can simply discard some rare incorrect CycleGAN segmentation results by quick human-inspection to prevent LithoNet from learning incorrect shape correspondences.

B. Step II: Shape Deformation Prediction

To learn the shape correspondence and the deformation field between SEM and layout images, LithoNet is trained on a collection of image pairs, each containing a layout and a ground-truth segmentation mask, i.e., a processed SEM image, generated by Step I described in Section III-A.

As shown in Fig. 3, LithoNet consists of a generator and a warping module. The generator is a U-net [33]-like network that outputs a 2-D dense correspondence map depicting the deformation field between the training image pairs. Then, using the sampling strategy used in the spatial transformer network (STN) [34], the warping module synthesizes a warped version of the given input layout to simulate wafer-fabricated circuitry based on the deformation map. STN is a differentiable module designed for enabling neural networks to actively spatially transform feature maps so that neural network models can learn invariance to translation, scale, rotation, and warping. Consequently, we adopt the sampling strategy of STN to benefit our LithoNet.

Moreover, the deformation map $\mathcal{M} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ describes the pixel-to-pixel displacement from a source layout image $\mathcal{S}$ to an SEM-styled image $\mathcal{J}$. Therefore, after LithoNet learns to predict the pixel-to-pixel correspondence, we apply the deformation map $\mathcal{M}$ on the layout $\mathcal{S}$ to derive the deformed shape contour. The warping process that relates $\mathcal{S}$, $\mathcal{J}$, and $\mathcal{M}$ can be expressed as $\mathcal{J}(m, n) = \mathcal{S}(\mathcal{M}^{-1}(m, n))$, where (m, n) denotes the pixel coordinate.

In contrast to common image generation networks like [18] and [35], the advantages of LithoNet are twofold. First, LithoNet can generate and visualize a predicted deformation field and, therefore, what has been learned by the network, i.e., the shape correspondences between input training image pairs, can be verified straightforwardly. Second, based on the visualized deformation field, it would be easier to identify possible impacts (e.g., defects), whether global or local, caused by the layout and the configuration parameters during fabrication process, on the physical appearance of an IC's metal layer. Concisely, the deformation field generated by our LithoNet is beneficial for clarifying both global and local shape correspondences between a layout and the SEM image of its product IC.

C. Training Loss Functions

The training loss function $\mathcal{L}_{\text{total}}$ of LithoNet is primarily defined in the following form:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{var}} + \mathcal{L}_{\text{smooth}} + \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{par}} \tag{1}$$

where $\mathcal{L}_{\text{rec}}$ denotes the reconstruction loss that measures the dissimilarity between the training ground-truth $\mathcal{I}$ and the synthetic SEM-styled image $\mathcal{J}$. Meanwhile, $\mathcal{L}_{\text{var}}$ measures the variability difference between a paired training image pair, and $\mathcal{L}_{\text{smooth}}$ guarantees the smoothness of the deformation map. Finally, $\mathcal{L}_{\text{reg}}$ is used to penalize large displacements on the deformation map, and $\mathcal{L}_{\text{par}}$ is the regression loss of fabrication parameters.

1) Reconstruction Loss: The reconstruction loss term $\mathcal{L}_{\text{rec}}(\mathcal{I}, \mathcal{J})$ is defined as the L_1-loss between the training ground-truth $\mathcal{I}$ and the synthetic SEM-styled image $\mathcal{J}$ as follows:

$$\mathcal{L}_{\text{rec}}(\mathcal{I}, \mathcal{J}) = \frac{1}{n}\|\mathcal{I} - \mathcal{J}\|_1 \tag{2}$$

where n denotes the number of pixels. We derive $\mathcal{L}_{\text{rec}}$ by the following steps: 1) densely sampling pixel positions on the to-be-generated $\mathcal{J}$; 2) locating the correspondences of them on the input layout according to the deformation map $\mathcal{M}$ that records the mapping relationship between pixels on $\mathcal{I}$ onto their counterparts on $\mathcal{J}$; 3) using backward interpolation to estimate the sampled pixel values on $\mathcal{J}$, i.e., $\hat{\mathcal{J}}(x, y) = \mathcal{I}(\hat{x}, \hat{y})$ with noninteger positions $(\hat{x}, \hat{y}) = \mathcal{M}^{-1}(x, y)$; and 4) generating an estimated $\hat{\mathcal{J}}$ via bilinear interpolation[1] to calculate $\mathcal{L}_{\text{rec}}$.

2) Total Variation Loss: The total variation loss $\mathcal{L}_{\text{var}}(\mathcal{I}, \mathcal{J})$ is defined as the total variation [36] of the *signed* difference between $\mathcal{I}$ and $\mathcal{J}$, that is

$$\mathcal{L}_{\text{var}}(\mathcal{I}, \mathcal{J}) = \sum |\nabla(\mathcal{I} - \mathcal{J})|. \tag{3}$$

This term is designed to align the shape contours of $\mathcal{J}$ with those of $\mathcal{I}$. Without this term, the loss function might be dominated by the reconstruction loss described in (2), and consequently LithoNet would generate a bizarre synthetic image $\mathcal{J}$, which can produce a high overlap ratio compared with ground-truth image $\mathcal{I}$ but has unnaturally jiggling contours. In other words, $\mathcal{L}_{\text{var}}$ aims to retain the shape similarity.

3) Smoothness Loss: The smoothness loss is a penalty term defined as the L_1-norm of the weighted gradient of the deformation map

$$\mathcal{L}_{\text{smooth}} = \|(\nabla \mathcal{M}) \circ \mathbf{W}\|_1 \tag{4}$$

where $\circ$ denotes the Hadamard product, and $\mathbf{W}$ is an edge-aware weighting matrix defined as

$$\mathbf{W}(x, y) = e^{-(|\nabla \mathcal{S}(x,y)| + |\nabla \mathcal{I}(x,y)|)}. \tag{5}$$

Note that contour edges on the input layout $\mathcal{S}$ and the ground-truth layout-styled SEM image $\mathcal{I}$ result in discontinuities in the deformation map $\mathcal{M}$. Because such discontinuities contribute to an unnecessary smoothness penalty, $\mathcal{L}_{\text{smooth}}$ should be suppressed appropriately according to the gradient information of both layout and SEM images.

4) Regularization Loss: The regularization loss is defined as the L_1-norm of deformation map $\mathcal{M}$

$$\mathcal{L}_{\text{reg}} = \|\mathcal{M}\|_1. \tag{6}$$

This term reflects the fact that the deformation caused by wafer fabrication tends to be small, as will be discussed in Section V-D2.

5) Regression Loss for Fabrication Parameters: Because the configuration parameters of a fabrication process are continuous variables that influence the physical appearance of the wafer layer, we formulate the relationship between the fabrication parameters and the appearance of wafer layer as a regression problem. The regression loss $\mathcal{L}_{\text{par}}$ is defined as

$$\mathcal{L}_{\text{par}} = \underbrace{\|D_y(G(\mathcal{S}|y)) - y\|_2}_{\text{Generator loss}} + \underbrace{\|D_y(\mathcal{I}_y) - y\|_2}_{\text{Discriminator loss}} \tag{7}$$

where $\mathcal{I}$ is the ground-truth shape segmented from the SEM image used for training; y is the fabrication parameter vector corresponding to $\mathcal{I}_y$; $\mathcal{S}$ denotes the input layout; and $G(S|y)$ is the predicted deformed shape. Therefore, this loss term aims to train: 1) a generator able to predict a synthesized SEM-styled image based on the given $\mathcal{S}$ and y and 2) a discriminator able to discriminate whether each entry of the extracted parameter vector $D_y(\mathcal{I}_y)$ is identical to the corresponding entry within the ground-truth fabrication parameter vector y.

IV. OPCNet: A CNN-Based Photomask Corrector

As described in Section II-C, the major challenge in developing a learning-based mask optimizer is to collect a comprehensive amount of ground-truth mask data, e.g., well OPC-corrected photomasks of various layout patterns, leading to desired shapes of fabricated circuitry. This is, however, very costly and time consuming. To overcome this difficulty, as shown in Fig. 2(b), we utilize a pretrained LithoNet as an auxiliary module to train our photomask optimizer, i.e., OPCNet. Given an IC layout pattern, OPCNet aims to predict an OPC-corrected mask pattern so that, after being deformed by the lithography and etching processes that are simulated by LithoNet, the predicted deformed shape will be as close as possible to the original layout pattern. By regarding the LithoNet–OPCNet network as a composite function $f = h \circ g$ with h and g denoting, respectively, LithoNet and OPCNet, this design can be expressed as min $\|\mathcal{S} - f(\mathcal{S})\|$, where $\mathcal{S}$ and $f(\mathcal{S})$ are, respectively, the input layout and the final prediction produced by the LithoNet–OPCNet network. Therefore, because such minimization optimizes to $f = 1$, which implies $h \circ g = 1$, OPCNet and LithoNet should be the inverse functions of each other. As a result, for a desired layout pattern, we can use the predicted output of OPCNet as the input of LithoNet, and the desired layout itself as the corresponding input of OPCNet. Consequently, we can train OPCNet without the need for collecting the "ground-truth" OPC-corrected photomasks.

Specifically, given a layout design pattern $\mathcal{S}$, OPCNet aims to generate a photomask $\mathcal{K}$, whose lithography and etching simulation result $\mathcal{J}$ predicted by LithoNet best matches $\mathcal{S}$. This design makes our OPCNet "ground-truth-free" during the training stage, assuming LithoNet is already well-trained. In addition, with the design of the *input–output consistency loss* used to measure the dissimilarity between a layout design pattern $\mathcal{S}$ and its lithography simulation result $\mathcal{J}$, OPCNet becomes a self-supervised learning method. The whole pipeline of our mask optimization method is illustrated in Fig. 2(b). Note that: 1) the pretrained LithoNet is fixed while training OPCNet and 2) OPCNet is intrinsically a generator for translating a layout pattern $\mathcal{S}$ into its optimal photomask $\mathcal{K}$ based on the wafer fabrication model learned by LithoNet.

A. Application Scenarios

The LithoNet–OPCNet network can serve two purposes. First, when LithoNet is well pretrained on a comprehensive set of (layout, SEM) pairs if during IC fabrication no OPC is performed, or on a set of (mask, SEM) pairs if OPC is performed, LithoNet can accurately predict the shape deformations due to the lithography and etch processes. Since OPCNet is the inverse function of LithoNet, it can be used to predict the OPC-optimized mask for a target layout pattern that would minimize the discrepancy between the fabricated IC shape and the target layout pattern without the need for collecting ground-truth OPC-optimized masks. In this way, the LithoNet–OPCNet network potentially can replace the function of current OPC prediction models.

[1] $\hat{\mathcal{J}}(x, y) = \mathcal{I}(\hat{x}, \hat{y}) \approx \begin{bmatrix} \lceil \hat{x} \rceil - \hat{x} \\ \hat{x} - \lfloor \hat{x} \rfloor \end{bmatrix}^t \begin{bmatrix} \mathcal{I}(\lfloor \hat{x} \rfloor, \lfloor \hat{y} \rfloor) & \mathcal{I}(\lfloor \hat{x} \rfloor, \lceil \hat{y} \rceil) \\ \mathcal{I}(\lceil \hat{x} \rceil, \lfloor \hat{y} \rfloor) & \mathcal{I}(\lceil \hat{x} \rceil, \lceil \hat{y} \rceil) \end{bmatrix} \begin{bmatrix} \lceil \hat{y} \rceil - \hat{y} \\ \hat{y} - \lfloor \hat{y} \rfloor \end{bmatrix}$, where $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote *ceiling* and *floor* functions, respectively.

Second, if the training samples are not comprehensive enough to train a fully reliable LithoNet model, the LithoNet–OPCNet may not be able to completely replace current OPC prediction models. However, if LithoNet can achieve a reasonable accuracy, the LithoNet–OPCNet network can still be used to verify if there is any inconsistency between the optimized mask prediction and the conventional OPC mask—an obvious inconsistency implies the fab-plant need to update the OPC model by collecting specific process-window conditions over the input layout structure.

B. Training Loss Functions for OPCNet

The overall training loss $\mathcal{L}_{\mathcal{K}}$ of OPCNet is defined as

$$\mathcal{L}_{\mathcal{K}} = \mathcal{L}_{IO} + \mathcal{L}_{K\text{var}} + \mathcal{L}_{K\text{smooth}} \tag{8}$$

where $\mathcal{L}_{IO}$ denotes the input–output consistency loss measuring the dissimilarity between input layout $\mathcal{S}$ and LithoNet's output $\mathcal{J}$, $\mathcal{L}_{K\text{var}}$ represents the total variation loss on the difference between $\mathcal{S}$ and $\mathcal{J}$, and $\mathcal{L}_{K\text{smooth}}$ denotes the mask smoothness loss for ensuring the smoothness of the obtained photomask patterns $\mathcal{K}$.

1) Input–Output Consistency Loss: The input–output consistency loss $\mathcal{L}_{IO}(\mathcal{S}, \mathcal{J})$ aims to guide the learning of OPCNet so that the shape predicted by LithoNet $\mathcal{J}$ best matches the desired input layout $\mathcal{S}$, provided that the source layout is OPC-corrected by the learned OPCNet. The loss term is defined as follows:

$$\mathcal{L}_{IO}(\mathcal{S}, \mathcal{J}) = \frac{1}{n}\|\mathcal{S} - \mathcal{J}\|_1 \tag{9}$$

where n denotes the number of pixels.

2) Total Variation Loss: Similar to (3), the total variation loss $\mathcal{L}_{K\text{var}}(\mathcal{S}, \mathcal{J})$ is defined as the total variation of signed difference between the input layout $\mathcal{S}$ and the prediction of LithoNet $\mathcal{J}$

$$\mathcal{L}_{K\text{var}}(\mathcal{S}, \mathcal{J}) = \sum |\nabla(\mathcal{S} - \mathcal{J})| \tag{10}$$

which is again an empirical term used to avoid unnatural patterns on the predicted shapes. $\mathcal{L}_{K\text{var}}$ prevents $\mathcal{L}_{\mathcal{K}}$ from being dominated by the I/O-consistency loss $\mathcal{L}_{IO}$. Without this term, the OPCNet may produce a unnatural correction.

3) Mask Smoothness Loss: The mask smoothness loss is defined to be the L_1-norm of the gradient of the mask prediction, that is

$$\mathcal{L}_{K\text{smooth}} = \|\nabla\mathcal{K}\|_1. \tag{11}$$

This term penalizes the discontinuity on the corrected photomask $\mathcal{K}$ to guarantee the smoothness of shape contours of $\mathcal{K}$. Note that $\mathcal{L}_{K\text{smooth}}$ does not incorporate with an edge-aware weighting matrix, since there are no ground-truth masks that define true contour edges in the training dataset.

In practice, there are some restrictions on what kind mask shapes can be made by a mask shop. We can integrate such mask manufacturing rules checking (MRC) with OPCNet in two ways: 1) formulating the MRC as training loss functions of OPCNet or 2) using a post-processing step based on the MRC rules to modify the OPC-corrected layout patterns generated by OPCNet. The second method is commonly used in practice, but OPCNet has the capability to adopt the first method or a combination of the two methods.

V. Experimental Results

A. Dataset and Settings

Images demonstrated in this work are selected from two datasets provided by United Microelectronics Corporation (UMC). These two UMC datasets consist of pairs of images, each containing one layout image patch and its wafer's SEM image patch. UMC dataset #1 contains SEM images taken from wafers fabricated with the same fabrication parameters, and UMC dataset #2 contains SEM images taken from wafers fabricated with *seven* various normalized parameter settings ranging from -0.9 to $+0.9$. In total, UMC dataset #1 contains: 1) a 928-pair training subset and 2) a 114-pair blind testing subset, whereas UMC dataset #2 contains: 1) a subset comprising 1057×7 pairs[2] for training and 2) another subset comprising 12×7 pairs for blind testing. All images in the blind testing set are collected from historical fabrication data. Compared with those in the training sets, the blind test images are of much larger sizes and contain unseen design patterns. We trained CycleGAN for style-transfer in Step I on the UMC dataset #1, and LithoNet on UMC datasets #1 and #2. As for OPCNet, it was trained on paired data, each of which contains: 1) a layout image $\mathcal{S}$ in the first dataset and 2) its fabricated IC shape $\mathcal{J}$ predicted by feeding $\mathcal{S}$ into a pretrained LithoNet. In our experiments, all image patches are downscaled from 512×512 to 256×256 to reduce the computational complexity. Each 512×512 source image corresponds to a 2×2 μm^2 region, so aliasing will not occur in this case. The five loss terms described in (1) are weighted empirically by (100, 0.001, 150, 0.002, 10). Meanwhile, the weighting coefficients for OPCNet described in (8) are (50, 0.001, 50). These weighting coefficients are determined according to the following two steps. First, because the reconstruction loss and the smoothness loss in (1) and (8) are more considerable than the others, we assign them with larger weighting coefficients and adjust the weighting coefficients until reaching reasonable results. In this step, the coefficients of other loss terms are temporarily set to be zero. Second, we assign the other loss terms with much smaller coefficients initially and then adjust them to make the training process easily converge.

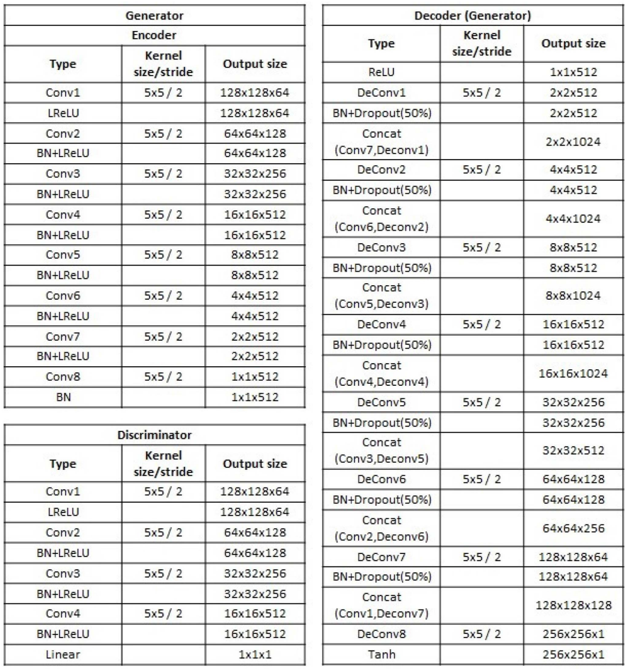
B. Architecture and Run-Time Information

Fig. 4 shows the architectures of subnetworks constituting LithoNet, including: 1) the encoder of the generator; 2) the decoder of the generator; and 3) the discriminator. OPCNet shares the same architecture as LithoNet's generator. On average, LithoNet and OPCNet take 0.0156 and 0.0150 s to run a simulation on a 256×256 image on an NVIDIA 2080Ti GPU, respectively. The whole training process takes about 1.5 days on a server equipped with one NVIDIA P100 GPU. Note that on the server, it takes about 34 s to run OPC contour simulation for a 4×1.7 μm^2 layout patch.

C. Performance Metrics

The performance of our model is evaluated objectively in terms of some widely used similarity metrics, including intersection over union (IOU), SSIM [38], and per pixel error

[2] There are 1057 layouts and 7 different settings per layout, so 7399 pairs of images in total.



Generator		
Encoder		
Type	Kernel size/stride	Output size
Conv1	5x5 / 2	128x128x64
LReLU		128x128x64
Conv2	5x5 / 2	64x64x128
BN+LReLU		64x64x128
Conv3	5x5 / 2	32x32x256
BN+LReLU		32x32x256
Conv4	5x5 / 2	16x16x512
BN+LReLU		16x16x512
Conv5	5x5 / 2	8x8x512
BN+LReLU		8x8x512
Conv6	5x5 / 2	4x4x512
BN+LReLU		4x4x512
Conv7	5x5 / 2	2x2x512
BN+LReLU		2x2x512
Conv8	5x5 / 2	1x1x512
BN		1x1x512

Discriminator		
Type	Kernel size/stride	Output size
Conv1	5x5 / 2	128x128x64
LReLU		128x128x64
Conv2	5x5 / 2	64x64x128
BN+LReLU		64x64x128
Conv3	5x5 / 2	32x32x256
BN+LReLU		32x32x256
Conv4	5x5 / 2	16x16x512
BN+LReLU		16x16x512
Linear		1x1x1

Decoder (Generator)		
Type	Kernel size/stride	Output size
ReLU		1x1x512
DeConv1	5x5 / 2	2x2x512
BN+Dropout(50%)		2x2x512
Concat (Conv7,Deconv1)		2x2x1024
DeConv2	5x5 / 2	4x4x512
BN+Dropout(50%)		4x4x512
Concat (Conv6,Deconv2)		4x4x1024
DeConv3	5x5 / 2	8x8x512
BN+Dropout(50%)		8x8x512
Concat (Conv5,Deconv3)		8x8x1024
DeConv4	5x5 / 2	16x16x512
BN+Dropout(50%)		16x16x512
Concat (Conv4,Deconv4)		16x16x1024
DeConv5	5x5 / 2	32x32x256
BN+Dropout(50%)		32x32x256
Concat (Conv3,Deconv5)		32x32x512
DeConv6	5x5 / 2	64x64x128
BN+Dropout(50%)		64x64x128
Concat (Conv2,Deconv6)		64x64x256
DeConv7	5x5 / 2	128x128x64
BN+Dropout(50%)		128x128x64
Concat (Conv1,Deconv7)		128x128x128
DeConv8	5x5 / 2	256x256x1
Tanh		256x256x1

Fig. 4. Network architecture of LithoNet. Its generator consists of an encoder and a decoder. OPCNet is architecturally identical to LithoNet's generator.

rate. These three metrics are defined below

$$\mathrm{IOU}(x, y) = \frac{\cap(x, y)}{\cup(x, y)} \tag{12}$$

$$\mathrm{ErrorRate} = \frac{\mathrm{FP} + \mathrm{FN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \text{and} \tag{13}$$

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C1)(2\sigma_{xy} + C2)}{(\mu_x^2 + \mu_y^2 + C1)(\sigma_x^2 + \sigma_y^2 + C2)} \tag{14}$$

where $\cap$ and $\cup$ denote, respectively, the set intersection and set union; and TP, TN, FP, and FN stand for true positive, true negative, false positive, and false negative, respectively. The SSIM index measures the structural similarity between two images. In the equation above, μ_x and σ_x denote the average and the variance of image x, σ_{xy} denotes the covariance, and C_1 and C_2 are variables stabilizing the division.

Finally, we also utilize the contour-to-contour distance, hereafter abbreviated as C2Cdist, to approximate the edge placement error (EPE) and the edge displacement error used in [2]. This metric, methodologically similar to EPE, measures the mean contour-to-contour distance between a lithography prediction and its SEM contour ground truth. We utilize this strategy because an SEM prediction usually contains multiple irregular regions whose bounding boxes may be overlapped, and thus bounding boxes cannot suggest a fair distance measure for the whole SEM prediction. The *C2Cdist* metric, measured in pixels, is illustrated in Fig. 5 and available for download at [39]. We will demonstrate in detail that our model outperforms other image-to-image translation methods and the standard OPC approach.

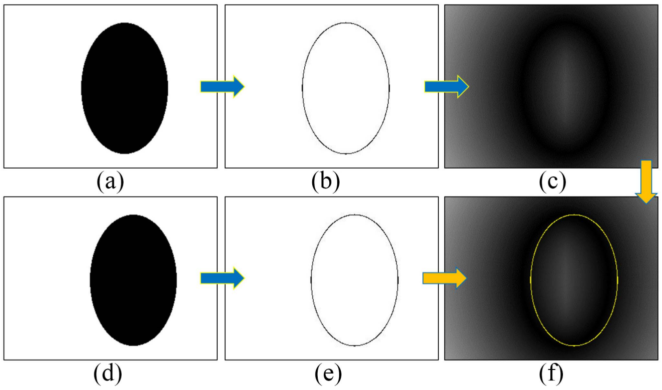


Fig. 5. Illustration of contour-to-contour distance (C2Cdist). (a) Ground truth (GT); (b) contour of GT; (c) distance map [37] of GT's contour obtained by MATLAB function *bwdist*; (d) input; (e) contour of the input; (f) overlay of (e) on GT's distance map. Then, *C2Cdist* can be derived by averaging distance values collected along the input's contour.

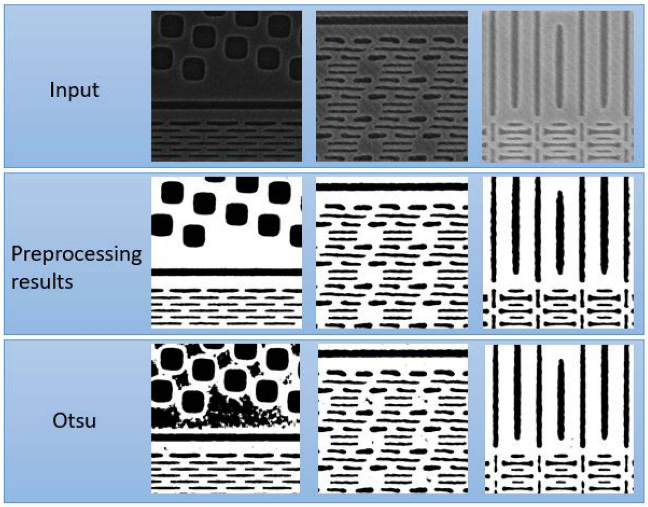


Fig. 6. Comparison between the segmentation masks obtained by CycleGAN [22] trained on the UMC dataset #1 and traditional Otsu thresholding.

D. LithoNet

1) Image Domain Transfer: In Fig. 6, we compare our image domain transfer results with images derived by the traditional Otsu thresholding method [29]. Obviously, the source SEM images contain typical complications from the SEM imaging process, such as bias in brightness/contrast probably due to gain-shift and scanning-pattern noise. It is thus difficult for common methods to threshold an SEM image appropriately. By exploiting a well-trained translator, e.g., CycleGAN [22], an SEM image can be transferred into a layout-styled format with its contour shapes unchanged.

2) Prediction Results: Fig. 7 illustrates the deformation map predicted from the input layout, the predictions of fabricated IC shapes based on the deformation map, and the corresponding ground truths of fabricated IC shapes extracted from their associated SEM images. The deformation maps show that LithoNet successfully learns to widen lines within open areas and to condense lines otherwise. Because such information is the key to the metrology applications, such as layout scoring and OPC simulation described in Fig. 1, this experiment also demonstrates that LithoNet can be used to bridge computer vision techniques with both fields of semiconductor manufacturing and computer-aided-design.

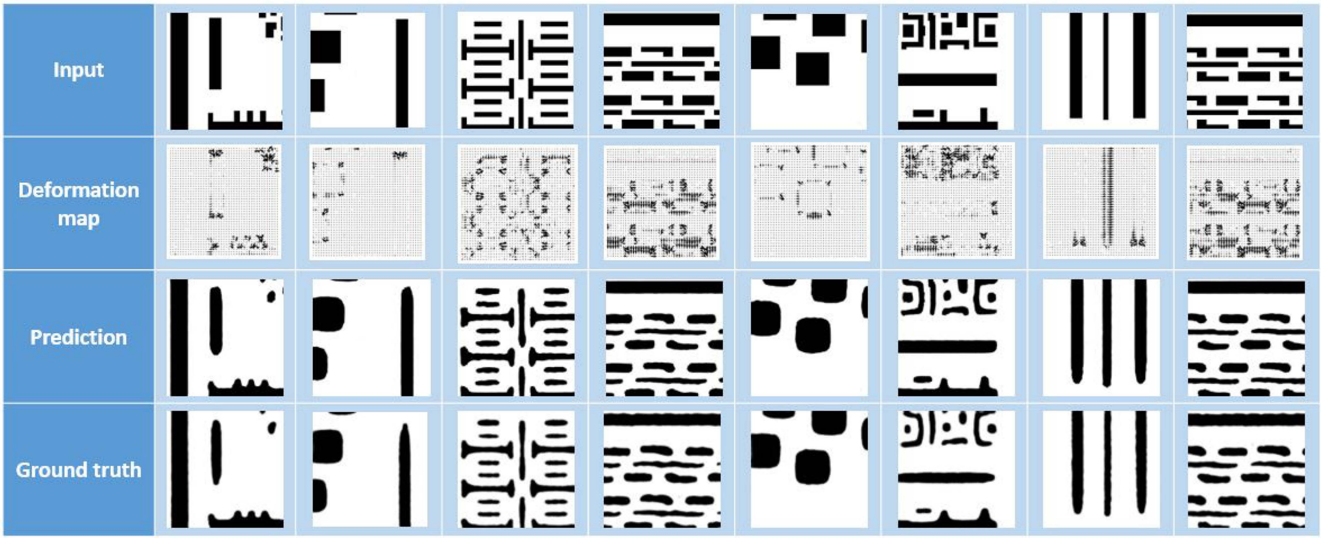


Fig. 7. Comparison of the input layout patterns, the predicted deformation maps, the predictions of fabricated IC shapes based on the deformation maps, and the ground truths of fabricated IC shapes extracted from their associated SEM images. The second row illustrates every deformation map $\mathcal{M}(m, n) = x_m\hat{i} + y_n\hat{j}$ as its per-pixel magnitude $\sqrt{x_m^2 + y_n^2}$ pointing to the deformation direction $\hat{v} = (x_m\hat{i} + y_n\hat{j})/\sqrt{x_m^2 + y_n^2}$.

TABLE I
ABLATION STUDY OF DIFFERENT LOSS SETTINGS ON THE UMC DATASET #1 (DATA IN PARENTHESES ARE FROM TRAINING SET)

Loss testing(training)	avg IOU	avg SSIM	avg Error	C2Cdist ($\bar{\mu} \pm \bar{\sigma}$)
Pix2pix	0.8868	0.8784	0.0361	0.3940±0.5156
	(0.9192)	(0.8553)	(0.0251)	(0.1891±0.3911)
$\mathcal{L}_{total}$	0.8846	0.8730	0.0371	0.4058±0.5240
-$\mathcal{L}_{var}$-$\mathcal{L}_{smooth}$	(0.9173)	(0.8605)	(0.0219)	(0.1838±0.3823)
$\mathcal{L}_{total}$-$\mathcal{L}_{var}$	0.8789	0.8658	0.0392	0.4295±0.5364
	(0.9130)	(0.8552)	(0.0262)	(0.2013±0.3982)
$\mathcal{L}_{total}$-$\mathcal{L}_{smooth}$	0.8849	0.8720	0.0368	0.4127±0.5312
	(0.9066)	(0.8563)	(0.0280)	(0.2414±0.4394)
$\mathcal{L}_{total}$	0.8820	0.8701	0.0380	0.4109±0.5192
(LithoNet)	(0.8847)	(0.9123)	(0.0362)	(0.3044±0.4653)

TABLE II
ABLATION STUDY OF DIFFERENT LOSS SETTINGS ON A SMALL SUBSET OF THE UMC DATASET #1

Loss testing(training)	avg IOU	avg SSIM	avg Error	C2Cdist ($\bar{\mu} \pm \bar{\sigma}$)
$\mathcal{L}_{total}$	0.8419	0.8109	0.1556	0.5782±0.7265
-$\mathcal{L}_{var}$-$\mathcal{L}_{smooth}$	(0.8724)	(0.8017)	(0.0864)	(0.2152±0.3876)
$\mathcal{L}_{total}$-$\mathcal{L}_{var}$	0.8462	0.8155	0.1502	0.5961±0.7388
	(0.8698)	(0.7972)	(0.0993)	(0.2427±0.4304)
$\mathcal{L}_{total}$-$\mathcal{L}_{smooth}$	0.8506	0.8223	0.1445	0.5823±0.7309
	(0.8653)	(0.7991)	(0.1036)	(0.2816±0.4584)
$\mathcal{L}_{total}$	0.8514	0.8208	0.1440	0.5797±0.7123
(LithoNet)	(0.8593)	(0.7986)	(0.1227)	(0.3247±0.4856)

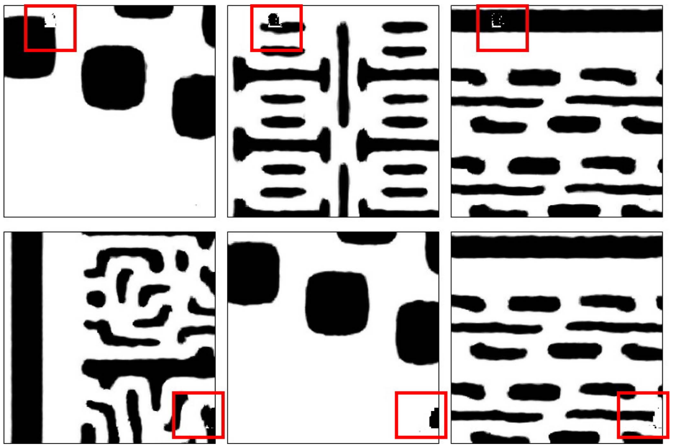


Fig. 8. Prediction results by LithoNet trained on the UMC dataset #1 without the smoothness loss term $\mathcal{L}_{\text{smooth}}$.

3) Ablation Study of Loss Terms: Here, we examine and discuss the effectiveness of individual loss terms in (1). First, we made numerical comparisons among different loss settings in Tables I and II, each of which corresponds to a different dataset. The values in parentheses are final loss values on the training set during training. The results shown in Table I were derived by LithoNet trained on the UMC dataset #1, whereas Table II shows the performance of LithoNet trained on a small subset of the UMC dataset #1 containing 480 training patches (obtained from 16 image samples by using only overlapped-cropping to obtain 30 patches for each sample for data augmentation). From Tables I and II, we can observe that the total-variation loss, $\mathcal{L}_{\text{var}}$, contributes significantly to the performance improvement. Moreover, $\mathcal{L}_{\text{smooth}}$ is beneficial to improve the objective performance when only a very limited amount of training samples is provided, as shown in Table II. On the contrary, as listed in Table I, $\mathcal{L}_{\text{smooth}}$ contributes less effectively to the objective performance when a comprehensive enough training dataset is given. We demonstrate the SEM-styled images predicted according to small training dataset without using the smoothness loss $\mathcal{L}_{\text{smooth}}$ in Fig. 8, where unexpected artifacts are highlighted in red rectangles. This experiment set shows the necessity of $\mathcal{L}_{\text{smooth}}$, especially in cases of a small training set.

The visual effect brought by the total-variation loss $\mathcal{L}_{\text{var}}$ is demonstrated in Fig. 9, where the "Baseline" column demonstrates images derived using $\mathcal{L}_{\text{total}} - \mathcal{L}_{\text{var}}$, whereas the "Full"

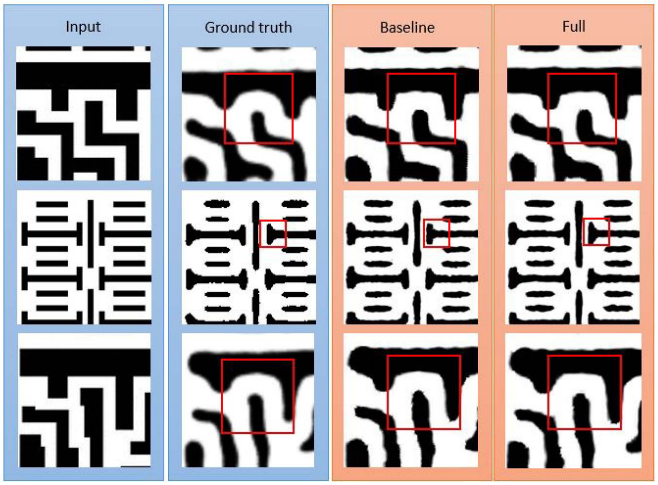


Fig. 9. Subject visual quality comparison of LithoNet with and without the total-variation loss $\mathcal{L}_{\text{var}}$, where the "Baseline" column demonstrates images derived using $\mathcal{L}_{\text{total}} - \mathcal{L}_{\text{var}}$ and the "Full" column shows predictions synthesized using $\mathcal{L}_{\text{total}}$.

column shows predictions synthesized using $\mathcal{L}_{\text{total}}$. This experiment set shows how $\mathcal{L}_{\text{var}}$ improves the visual quality of synthetic SEM-styled images. Take regions highlighted by red rectangles in Fig. 9 for example. Without $\mathcal{L}_{\text{var}}$, LithoNet tends to produce straight-line edges and sharp corners, although there are no such patterns on the training images produced by a real IC fabrication process, as shown in "Ground truth" column. By adding $\mathcal{L}_{\text{var}}$ to the total loss function, such artifacts can be largely mitigated, thereby more faithfully predicting the shapes of SEM images. Finally, note that LithoNet's $\mathcal{L}_{\text{var}}$ and $\mathcal{L}_{\text{reg}}$ can be regarded as regularization terms to prevent overfitting. As listed in Tables I and II, when LithoNet was trained on $\mathcal{L}_{\text{total}}$, its testing performance is close to that of the training data; and, such situation may not hold for other settings, including Pix2pix.

4) Comparison With Pix2pix: As LithoNet is a kind of image-to-image translation scheme, we compare it with Pix2Pix [18], a representative GAN-based image-to-image translation method. This experiment set was designed for two purposes. One is to verify if LithoNet is able to learn special shape correspondences between layout and SEM images, and the other is to check if LithoNet is more advantageous than Pix2Pix in this regard.

As shown in Table I, Pix2pix achieves slightly higher objective metric values than LithoNet. This situation, however, arises from the fact that these objective metrics mainly reflect the effect of the reconstruction loss term. Nevertheless, compared to Pix2pix, our total loss function described in (1) contains several additional loss terms, including $\mathcal{L}_{\text{reg}}$, $\mathcal{L}_{\text{par}}$, and $\mathcal{L}_{\text{smooth}}$, which do actually lead to better visual quality as will be explained later.

As illustrated in Fig. 10, Pix2pix produces artifacts like blurred and jiggled contour edges, whereas LithoNet is able to generate clear and smooth ones. Since both Pix2pix and LithoNet utilize L_1-norm to guarantee a global shape similarity, this phenomenon would probably be due to the different control strategies over local shapes. Specifically, LithoNet makes use of the total-variation loss, smoothness loss, and regularization loss to control the local deformations, whereas Pix2pix relies on its discriminator architecture, the so-called PatchGAN design that penalizes a structure at the scale of patches, to handle local deformations. Consequently, because PatchGAN does not put any penalty on blurred and jiggled edges and learns only to classify if each generated patch looks realistic, such artifacts are reasonable tradeoffs of Pix2pix's PatchGAN design.

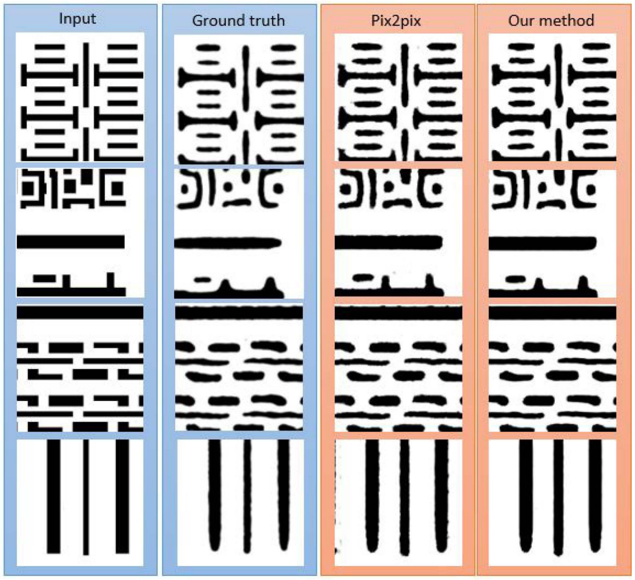


Fig. 10. Subjective visual quality comparison between Pix2pix and LithoNet, both trained on the UMC dataset #1.

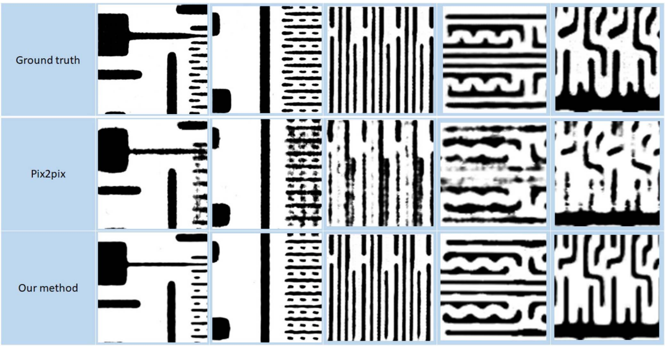


Fig. 11. Subjective visual quality comparison between Pix2pix and LithoNet, both trained on the UMC dataset #1, for some unseen layout patterns of a different observation scale.

Fig. 11 compares the prediction results of feeding LithoNet and Pix2pix with test images containing significantly distinct layout patterns from those in the training image set. Moreover, the source dimension of these testing images is much larger than the training data. Therefore, through this experiment, we can appraise the reliability and robustness of LithoNet and Pix2pix in mimicking an IC fabrication process when the input layout is a brand new, unseen pattern of a different scale. We can observe from Fig. 11 that, for unseen layout patterns of a different scale, LithoNet significantly outperforms Pix2pix in terms of the clarity and integrity of shape boundaries, although the predictions of LithoNet still cannot perfectly match the ground truth for lack of suitable training samples. Finally,

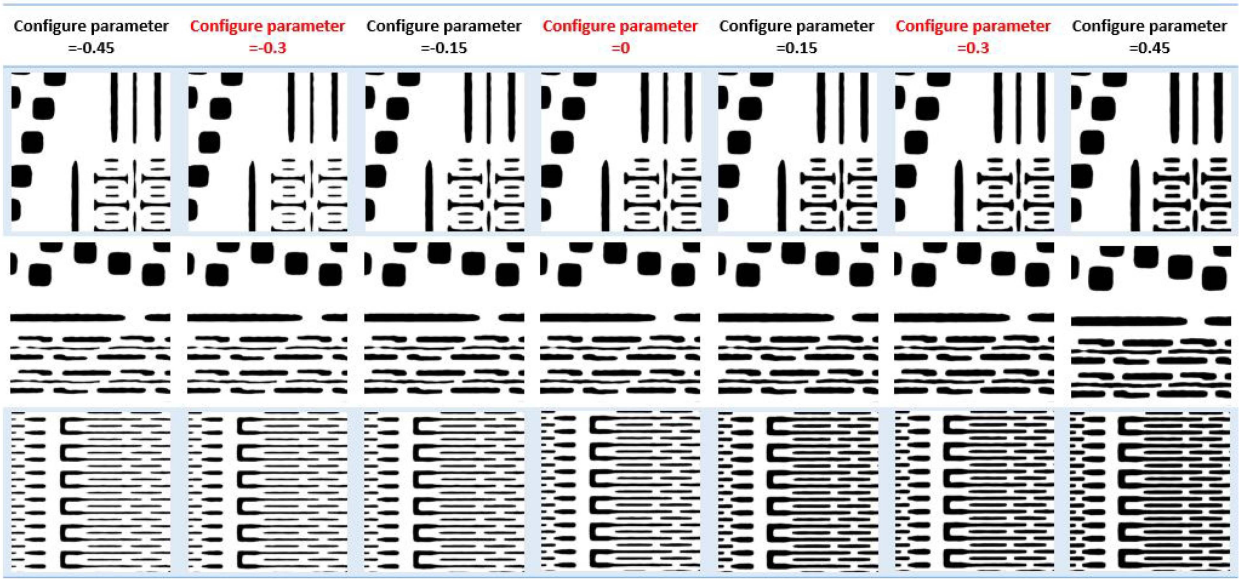

Configure parameter =-0.45
Configure parameter =-0.3
Configure parameter =-0.15
Configure parameter =0
Configure parameter =0.15
Configure parameter =0.3
Configure parameter =0.45

Fig. 12. Predictions by LithoNet trained on the UMC dataset #2 driven by different configuration parameter values for wafer fabrication. We focus on one configuration parameter which is inversely proportional to the degree of etching: the larger the parameter value, the lower the degree of etching, and the wider the metal lines. Those parameters values used in the training dataset are colored black, whereas those values not used in training are colored red.

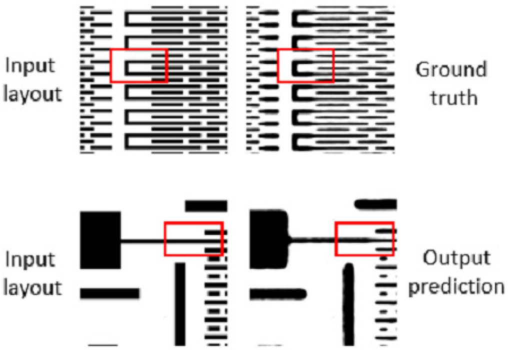
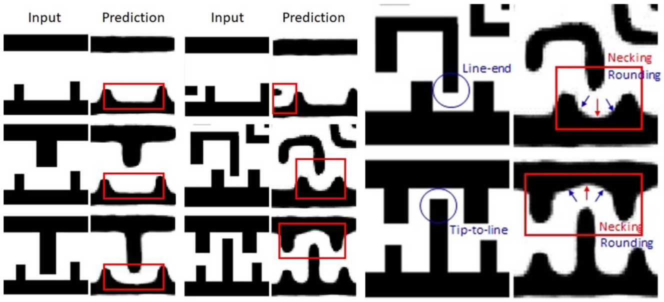


Fig. 13. Illustrations of inter-relationship between the shapes of metal lines and their local neighborhood.



(a) (b)

Fig. 14. Prediction results of LithoNet: (a) comparison between a layout and the prediction based on the layout and (b) conceptual illustration of "Necking" and "Rounding" where the necking effects are highlighted by red boxes and arrows and the rounding effects are indicated by blue arrows.

Table III lists the numerical comparisons between LithoNet and Pix2pix for this case.

Note that there is still no widely accepted objective metric to assess the quality of a predicted SEM-styled contour for an IC layout patch with respect to its SEM ground truth. Some conventional metrics, e.g., IOU and SSIM, measure the similarity globally but ignore local shape discrepancies which may lead to significant impact on IC manufacturability, whereas others, e.g., EPE and EDE [2], though designed for shape comparison, still cannot capture local shape discrepancies well. We here leave the problem of developing metrics capable of characterizing both local and global discrepancies and measuring the manufacturablity of a layout pattern simultaneously and as an open problem for future research.

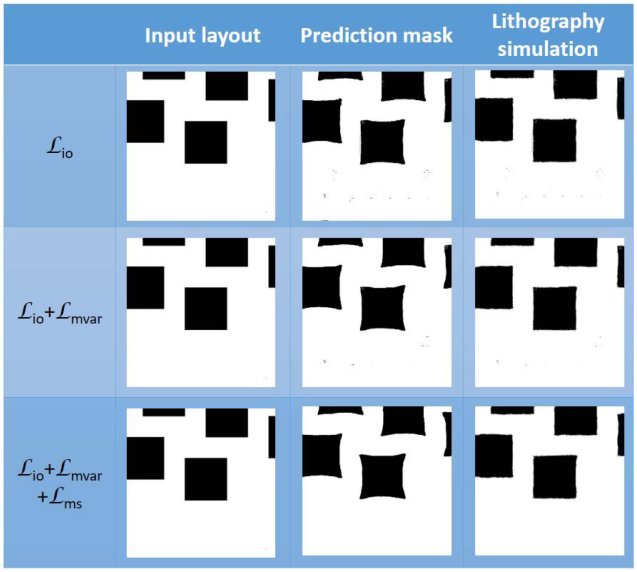

Input layout
Prediction mask
Lithography simulation
$\mathcal{L}_{\text{io}}$
$\mathcal{L}_{\text{io}}+\mathcal{L}_{\text{mvar}}$
$\mathcal{L}_{\text{io}}+\mathcal{L}_{\text{mvar}}+\mathcal{L}_{\text{ms}}$

Fig. 15. Illustrations of masks predicted by the mask generator and their lithography simulation outputs. The mean C2Cdist values between layout and lithography simulation of these three cases (from top to bottom) are 10.71, 5.50, and 0.34; and, the standard deviations are 22.73, 16.99, and 0.58.

5) *Fabrication Parameters:* Fig. 12 compares the predictions by LithoNet trained on the UMC dataset #2

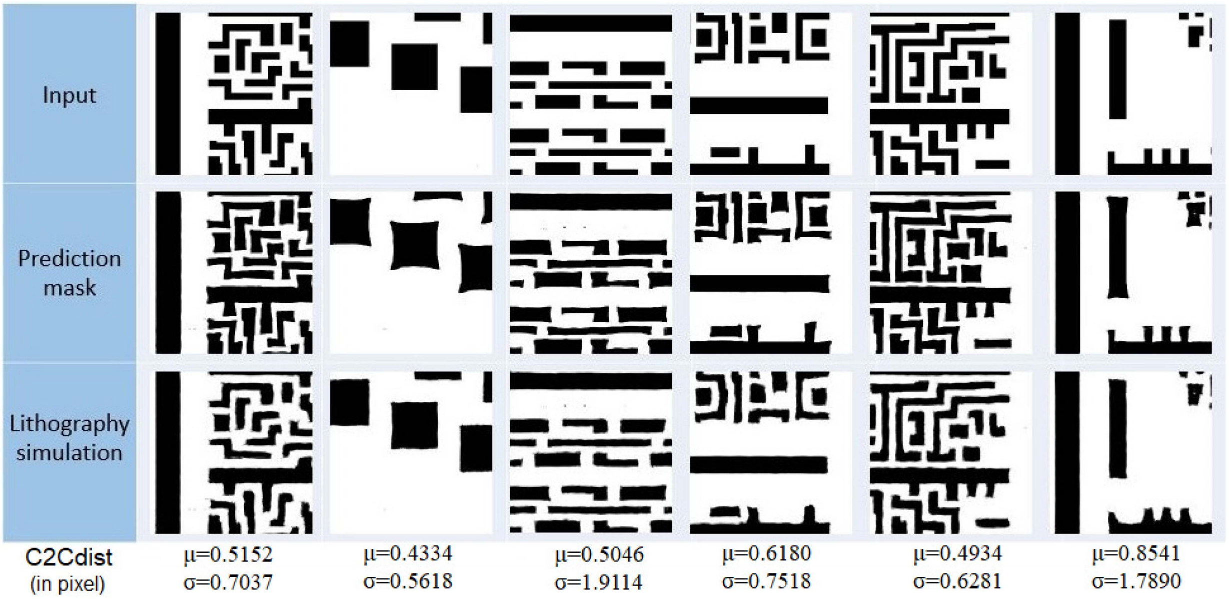


Fig. 16. Input layout $\mathcal{S}$, predicted mask $\mathcal{K}$, lithography simulation $\mathcal{J}$, and the *C2Cdist*($\mathcal{S}$, $\mathcal{J}$) value.

TABLE III
COMPARISON BETWEEN LITHONET AND PIX2PIX, BOTH TRAINED ON THE UMC DATASET #1, FOR UNSEEN LAYOUT PATTERNS OF A DIFFERENT SCALE

Method	avg IOU	avg SSIM	avg Error	C2Cdist ($\bar{\mu} \pm \bar{\sigma}$)
Pix2pix	0.6587	0.6396	0.1358	0.8179±0.7093
LithoNet	0.7107	0.6906	0.1170	0.8010±0.7080

driven by different configuration parameter values for wafer fabrication.

In this experiment set, we fix the *focus* and adjust the *energy* strength of the scanner in the lithography process to obtain the training samples, and then train LithoNet on them. We focus on one configuration parameter, i.e., *energy*, normalized to the range of $[-0.9, 0.9]$. This parameter is inversely proportional to the degree of etching: the larger the parameter value, the lower the degree of etching. This experiment set shows that LithoNet is capable of predicting the width of metal wires by using regression and the discriminator.

Those parameter values used in the training dataset are colored black, and those values not used in training are colored red. This experiment shows that the proposed LithoNet, thank to the regression loss term $\mathcal{L}_{\text{par}}$ described in (7), does learn the relationship between the line width and the fabrication parameter used to control the degree of etching in the fabrication process. Concisely speaking, the larger the parameter is, the wider the metal line should be. Hence, our LithoNet model is able to mimic the fabrication process and generate parameter-dependent prediction results. This is an important aspect of LithoNet design, and such design makes LithoNet suitable for semiconductor manufacturing simulations.

6) Model Generality: Here, we examine LithoNet's range of applicability. The image pair in the top row of Fig. 13 shows that, in an open area, the general fabrication process typically produces a metal line wider than its layout design, as highlighted by the red rectangle. The predicted image shown in the bottom row of Fig. 13 demonstrates that LithoNet learns the shape correspondence between paired training images, so it predicts a wider line in an open area and a narrower one in between two neighboring lines. Consequently, LithoNet can be expected to forecast fabrication results as long as a large enough amount of training data is given.

We also design another experiment to show that LithoNet can learn the "necking" and "rounding" effects that usually occur in IC fabrication, as highlighted by red rectangles in Fig. 14(a) and indicated by the red and blue arrows in Fig. 14(b). Necking is a high-risk pattern caused by either a tip-to-line or a line-end too close to another line on the layout design. As illustrated in Fig. 14(b), such situations may result in a line narrower than designed after fabrication. Hence, this experiment set provides further evidence that a well-trained LithoNet is capable of mimicking the semiconductor lithography and etch processes.

E. OPCNet

1) Impacts of Loss Functions: As described in Section IV, given a layout design pattern $\mathcal{S}$, OPCNet aims to generate a mask $\mathcal{K}$ whose lithography simulation result $\mathcal{J}$ predicted by LithoNet is most similar to $\mathcal{S}$. OPCNet is controlled jointly by the IO-consistency loss $\mathcal{L}_{IO}$, the total-variation loss $\mathcal{L}_{K\text{var}}$, and the mask smoothness loss $\mathcal{L}_{K\text{smooth}}$. The former two loss terms measure the dissimilarity between $\mathcal{S}$ and $\mathcal{J}$, and the third focuses on the smoothness of $\mathcal{K}$. Here, we examine how $\mathcal{L}_{K\text{var}}$ and $\mathcal{L}_{K\text{smooth}}$ contribute to the mask prediction task.

Shown in Fig. 15 are three columns of images, each of which corresponds to one loss setting. Comparing the mask predicted by using $\mathcal{L}_{IO}$ with that by $\mathcal{L}_{IO} + \mathcal{L}_{K\text{var}}$, we can find that $\mathcal{L}_{K\text{var}}$ guarantees the quality of shape contour in the lithography simulation. No matter the $\mathcal{L}_{\text{var}}$ of LithoNet or the $\mathcal{L}_{K\text{var}}$ of OPCNet, such total variation loss accounts for the difference between predicted contours and their ground truth and focuses on k pixels around the contour pixels. This term helps

$\mathcal{L}_{IO}$ to guarantee the similarity between the input layout and the lithography simulation and also avoid unexpected artifacts at contours. Finally, comparing the mask predicted by $\mathcal{L}_{IO} + \mathcal{L}_{K\mathrm{var}}$ with that by $\mathcal{L}_{IO} + \mathcal{L}_{K\mathrm{var}} + \mathcal{L}_{K\mathrm{smooth}}$, we find that $\mathcal{L}_{K\mathrm{smooth}}$ can globally suppress unexpected artifacts on the predicted mask image. The mask prediction derived by $\mathcal{L}_{mask}$ described in (8) can thus be artifact-free and smooth.

2) Mask Prediction Results: Finally, demonstrated in Fig. 16 are the masks predicted by OPCNet. Given a well-trained and accurate lithography simulator LithoNet, Fig. 16 provides evidence that OPCNet successfully performs the mask optimization task in a self-supervised learning manner without the need of collecting ground-truth OPC-corrected masks. With OPCNet, a layout pattern can be adequately corrected so that the resulting circuit shape best matches the source layout pattern, after an IC-fabrication process.

VI. Conclusion

In this article, we proposed a data-driven framework involving two CNNs: LithoNet and OPCNet. First, by learning the shape correspondence between paired training images, i.e., IC layout designs and their fabricated IC SEM images, LithoNet can predict the shape deformation field of the layout and then generate a lithography simulation result. Second, with pretrained LithoNet, OPCNet can learn a mask optimization model without ground-truth OPC-corrected masks based on the proposed input–output consistency loss. Experimental results evidently demonstrate that, in the lithography simulation issue, our method outperforms existing image-to-image translation schemes and the standard compact model-based simulations. In the mask optimization problem, OPCNet can correctly predict the mask whose lithography simulation image matches the expected layout. One on-going extension of this work is to establish a scoring system, based on the deformation map or SEM-styled image derived by our method, so that a VM system for IC circuit layout quality assessment can be developed.

References

[1] Y. Watanabe, T. Kimura, T. Matsunawa, and S. Nojima, "Accurate lithography simulation model based on convolutional neural networks," in *Proc. Opt. Microlithography XXX*, vol. 10147, 2017, Art. no. 101470K.

[2] W. Ye, M. B. Alawieh, Y. Lin, and D. Z. Pan, "LithoGAN: End-to-end lithography modeling with generative adversarial networks," in *Proc. 56th ACM/IEEE Design Autom. Conf.*, Las Vegas, NV, USA, 2019, pp. 1–6.

[3] A. Taflove and S. C. Hagness, *Computational Electrodynamics: The Finite-Difference Time-Domain Method*. Boston, MA, USA: Artech house, 2005.

[4] K. D. Lucas, H. Tanabe, and A. J. Strojwas, "Efficient and rigorous three-dimensional model for optical lithography simulation," *J. Opt. Soc. Amer. A*, vol. 13, no. 11, pp. 2187–2199, 1996.

[5] O. Otto *et al.*, "Automated optical proximity correction: A rules-based approach," in *Proc. Opt./Laser Microlithography VII*, vol. 2197, 1994, pp. 278–294.

[6] T.-J. Hsu, "Optical proximity correction (OPC) method for improving lithography process window," U.S. Patent 6 194 104, Feb. 2001.

[7] *Synopsys Inc*. Accessed: Aug. 25, 2020. [Online]. Available: https://www.synopsys.com/

[8] K. Aberman, J. Liao, M. Shi, D. Lischinski, B. Chen, and D. Cohen-Or, "Neural best-buddies: Sparse cross-domain correspondence," *ACM Trans. Graph.*, vol. 37, no. 4, p. 69, 2018.

[9] T. Zhou, P. Krahenbuhl, M. Aubry, Q. Huang, and A. A. Efros, "Learning dense correspondence via 3D-guided cycle consistency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, 2016, pp. 117–126.

[10] M.-H. Hung, T.-H. Lin, F.-T. Cheng, and R.-C. Lin, "A novel virtual metrology scheme for predicting CVD thickness in semiconductor manufacturing," *IEEE/ASME Trans. Mechatronics*, vol. 12, no. 3, pp. 308–316, Jun. 2007.

[11] G. A. Susto, S. Pampuri, A. Schirru, A. Beghi, and G. De Nicolao, "Multi-step virtual metrology for semiconductor manufacturing: A multilevel and regularization methods-based approach," *Comput. Oper. Res.*, vol. 53, pp. 328–337, Jan. 2015.

[12] A. Poonawala and P. Milanfar, "Mask design for optical microlithography—An inverse imaging problem," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 774–788, Mar. 2007.

[13] D. Z. Pan, B. Yu, and J.-R. Gao, "Design for manufacturing with emerging nanolithography," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 10, pp. 1453–1472, Oct. 2013.

[14] A. B. Kahng, "Reducing time and effort in IC implementation: A roadmap of challenges and solutions," in *Proc. ACM/ESDA/IEEE Design Autom. Conf.*, 2018, pp. 1–6.

[15] H. Yang, S. Li, Y. Ma, B. Yu, and E. F. Young, "GAN-OPC: Mask optimization with lithography-guided generative adversarial nets," in *Proc. 55th ACM/ESDA/IEEE Design Autom. Conf.*, San Francisco, CA, USA, 2018, pp. 1–6.

[16] B.-Y. Yu, Y. Zhong, S.-Y. Fang, and H.-F. Kuo, "Deep learning-based framework for comprehensive mask optimization," in *Proc. 24th Asia South Pac. Design Autom. Conf.*, 2019, pp. 311–316.

[17] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[18] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, 2017, pp. 1125–1134.

[19] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional s," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 8798–8807.

[20] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 700–708.

[21] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2013. [Online]. Available: arXiv:1312.6114.

[22] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Venice, Italy, 2017, pp. 2223–2232.

[23] Z. Yi, H. Zhang, P. Tan, and M. Gong, "DualGAN: Unsupervised dual learning for image-to-image translation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Venice, Italy, 2017, pp. 2849–2857.

[24] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, 2017, pp. 3722–3731.

[25] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 172–189.

[26] J.-R. Gao, X. Xu, B. Yu, and D. Pan, "MOSAIC: Mask optimizing solution with process window aware inverse correction," in *Proc. 51st ACM/EDAC/IEEE Design Autom. Conf.*, San Francisco, CA, USA, 2014, pp. 1–6.

[27] A. H. Gabor *et al.*, "Subresolution assist feature implementation for high-performance logic gate-level lithography," in *Proc. Opt. Microlithography XV*, vol. 4691, 2002, pp. 418–426.

[28] P. K. Saha and J. K. Udupa, "Optimum image thresholding via class uncertainty and region homogeneity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 7, pp. 689–706, Jul. 2001.

[29] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. 9, no. 1, pp. 62–66, Jan. 1979.

[30] K.-K. Maninis, S. Caelles, J. Pont-Tuset, and L. Van Gool, "Deep extreme cut: From extreme points to object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 616–625.

[31] G. Wang *et al.*, "DeepIGeoS: A deep interactive geodesic framework for medical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1559–1572, Jul. 2018.

[32] O. Barnich and M. Van Droogenbroeck, "ViBe: A universal background subtraction algorithm for video sequences," *IEEE Trans. Image Process.*, vol. 20, no. 6, pp. 1709–1724, Jun. 2011.

[33] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Medical Image Comput. Comput.-Assist. Intervent. (MICCAI)*, 2015, pp. 234–241.

[34] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.

[35] C. Wang, H. Zheng, Z. Yu, Z. Zheng, Z. Gu, and B. Zheng, "Discriminative region proposal adversarial networks for high-quality image-to-image translation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 770–785.

[36] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D, Nonlinear Phenomena*, vol. 60, nos. 1–4, pp. 259–268, 1992.

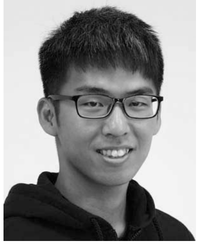
[37] C. Maurer, R. Qi, and V. Raghavan, "A linear time algorithm for computing exact euclidean distance transforms of binary images in arbitrary dimensions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 265–270, Feb. 2003.

[38] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[39] H.-C. Shao. (2020). *Contour-To-Contour Distance*. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/75551-contour-to-contour-distance

Hao-Chiang Shao (Member, IEEE) received the Ph.D. degree in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan, in 2012.

Since 2018, he has been an Assistant Professor with the Department of Statistics and Information Science, Fu Jen Catholic University, New Taipei City, Taiwan. From 2012 to 2017, he was a Postdoctoral Researcher with the Institute of Information Science, Academia Sinica, Taipei, Taiwan, where he was involved in a series of *Drosophila* brain research projects; from 2017 to 2018, he was an Research and Development Engineer with the Computational Intelligence Technology Center, Industrial Technology Research Institute, Hsinchu, taking charges of DNN-based automated optical inspection projects. His research interests include 2D+Z image atlasing, 3-D mesh processing, big industrial image data analysis, and machine learning.

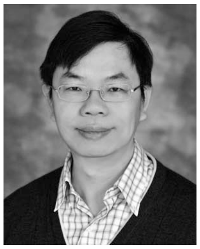


Chao-Yi Peng received the B.S. degree in electrical engineering from National Chung Cheng University, Minxiong, Taiwan, in 2017, and the M.S. degree in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan, in 2019.

Since 2019, he has been working with Altek Company, Torrington, CT, USA, as a Software Engineer. His research interests lie in computer vision, machine learning, and visual analytics for IC design for manufacturability.

Jun-Rei Wu received the B.S. degree in engineering science and ocean engineering from National Taiwan University, Taipei, Taiwan, in 2015, and the M.S. degree in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan, in 2019.

He is currently working as a Software Engineer with HTC VIVE. His research interests lie in computer vision, machine learning, and visual analytics for IC design for manufacturability.



Chia-Wen Lin (Fellow, IEEE) received the Ph.D. degree from National Tsing Hua University (NTHU), Hsinchu, Taiwan, in 2000.

He is currently a Professor with the Department of Electrical Engineering, and the Institute of Communications Engineering, NTHU. His research interests include image/video processing, computer vision, and machine learning.

Dr. Lin was a recipient of the Outstanding Electrical Engineer Professor Award presented by the Chinese Institute of Electrical Engineering, Taiwan, and two best paper awards from VCIP 2010 and 2015. He has served as an Associate Editor of the IEEE Transactions on Image Processing, the IEEE Transactions on Circuits and Systems for Video Technology, the IEEE Transactions on Multimedia, and IEEE Multimedia. He served as a Distinguished Lecturer of IEEE Circuits and Systems Society from 2018 to 2019. He is the Chair of IEEE ICME Steering Committee. He served as the TPC Co-Chair of IEEE ICIP 2019 and IEEE ICME 2010, and the General Co-Chair of IEEE VCIP 2018. He served as a Steering Committee Member of the IEEE Transactions on Multimedia from 2013 to 2015.

Shao-Yun Fang (Member, IEEE) received the B.S. degree in electrical engineering from National Taiwan University (NTU), Taipei, Taiwan, in 2008, and the Ph.D. degree from the Graduate Institute of Electronics Engineering, NTU, in 2013.

She is currently an Associate Professor with the Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei. Her current research interests focus on physical design and design for manufacturability for integrated circuits.

Dr. Fang was a recipient of two Best Paper Awards from the 2016 International Conference on Computer Design and the 2016 International Symposium on VLSI Design, Automation, and Test, and two Best Paper Nominations from the 2012 and 2013 International Symposium on Physical Design.

Pin-Yian Tsai received the M.S. degree in physics from National Tsing Hua University, Hsinchu, Taiwan, in 2008.

He is currently a Technical Manager of the Product Engineering Department, United Microelectronics Corporation (UMC), Hsinchu. He participated in advanced node product ramp-up and is currently working and researching on the field of Design for Manufacturing. He is currently focusing on developing methods for predicting weak patterns in layout manufacturing and automatic optical proximity correction to improve the manufacturing yield.

Yan-Hsiu Liu received the M.S. degree in chemistry from National Tsing Hua University, Hsinchu, Taiwan, in 2002.

In 2004, he joined United Microelectronics Corporation, Hsinchu, as a Process Integration Engineer, where he is currently working as a Deputy Department Manager on the development of smart manufacturing and responsible for industry-academia cooperation/collaboration. His research interests include the areas of intelligent manufacturing systems, adaptive parameter estimation, and neural networks.