

Machine Learning Algorithms and Architectures for SEM Based Defect  
Inspection and Metrology Challenges in Semiconductor Manufacturing

A Dissertation  
Presented to the  
Graduate Faculty of the  
University of Louisiana at Lafayette  
In Partial Fulfillment of the  
Requirements for the Degree of  
Doctor of Philosophy

Bappaditya Dey

Spring 2022

© Bappaditya Dey

2022

All Rights Reserved

Machine Learning Algorithms and Architectures for SEM Based Defect  
Inspection and Metrology Challenges in Semiconductor Manufacturing

Bappaditya Dey

**APPROVED:**

Magdy A. Bayoumi, Chair  
Head and Professor  
Department of Electrical and Computer Engineering

Ashok Kumar  
Associate Professor  
School of Computing and Informatics

Michael W. Totaro  
Associate Professor  
School of Computing and Informatics

Sandip Halder  
Group Leader, Advanced Patterning Department  
IMEC

Dr. Mary Farmer-Kaiser  
Dean of the Graduate School

*To my parents and my family.*

*Tama Dey, Bimal Chandra Dey, Swagata Mukherjee, Brishtilekha Dey,  
Sachin Bishu and Swapna Bishu*

## Acknowledgments

*“By the grace of the Guru, a disciple becomes a Pandit (scholar) even without reading books.”*

-- Swami Vivekananda

First and foremost, I would like to express my sincere gratitude to my esteemed advisor Dr. Magdy Bayoumi for his invaluable supervision, support, and encouragement throughout my graduate studies at the Center for Advanced Computer Studies (CACS), University of Louisiana at Lafayette. I found myself fortunate enough to have an advisor like him. My gratitude also extends to Dr. Sandip Halder for his encouragement towards the research as well as contributing to many of the research ideas, guiding me with patience, critiquing my experiment methods and results and providing significant feedback on my manuscripts. I will also take the opportunity to thank my committee members, Dr. Ashok Kumar, and Dr. Michael W. Totaro for their valuable suggestions and comments.

My special thanks go to my friend Dr. Kasem Khalil, without whose support I couldn't have achieved this.

I want to thank Mr. Kaushik Sah, Dr. Luka Kljucar, Dr. Sayantan Das, Mr. Dorin Cerbu, Dr. Gian Lorusso and other colleagues from imec, Leuven, Belgium for providing data to conduct my experiments as well as for providing insights and expertise that helped my experimental work.

I would like to thank my friends, colleagues, the department faculties, and university staff who made my life at University of Louisiana at Lafayette unforgettable.

Finally, my appreciation goes out to my parents for their unconditional love, prayers, and support throughout my whole life. Undoubtedly, it was not possible to achieve this goal without many of their sacrifices. Therefore, it is an honour for me to dedicate this work to them.

## **Contributors and Funding Sources**

### **Contributors**

This work was supported by a dissertation committee consisting of Professor Magdy A. Bayoumi, serving as the principal advisor, Dr. Sandip Halder from imec, Leuven, Belgium, Professor Ashok Kumar, Associate Dean of the Ray P. Authement College of Sciences, and Professor Michael W. Totaro, Associate Director (Operations) School of Computing and Informatics.

The student used the imec Leuven research computing facility and data to conduct the research. Abstract and/or baseline deep learning frameworks, libraries, and software, used in this research work, are cloned from corresponding official open-source GitHub repositories acknowledging proper credit to the contributors.

All other works conducted for the dissertation was completed by the student independently.

### **Funding Sources**

My graduate studies were supported by a research assistantship from Dr. Magdy A. Bayoumi and a teaching assistantship from the Center for Advanced Computer Studies (CACS) at University of Louisiana at Lafayette. My graduate studies were supported by a research assistantship from Dr. Danella Zhao from 2015-2016.

## Table of Contents

<b>Dedication .....</b>	iv
<b>Acknowledgments .....</b>	v
<b>Contributors and Funding sources .....</b>	vi
<b>List of Tables .....</b>	ix
<b>List of Figures .....</b>	x
<b>Abbreviations.....</b>	xvii
<b>Chapter 1: Introduction .....</b>	1
<b>1.1. Role of Lithography in Semiconductor Manufacturing .....</b>	3
<b>1.1.1. Optical lithography .....</b>	3
<b>1.1.2. Basic lithography equipment .....</b>	5
<b>1.2. Critical Dimension (CD): Size of the Structures we Print .....</b>	6
<b>1.3. Rayleigh Equation .....</b>	9
<b>1.4. Process Window .....</b>	9
<b>1.5. Wavelengths and Light Sources Used in Lithography .....</b>	11
<b>1.6. Double Patterning .....</b>	12
<b>1.7. Importance of High NA and Practical Challenges .....</b>	14
<b>1.7.1. High NA and depth-of-focus .....</b>	16
<b>1.7.2. High NA and resist thickness .....</b>	16
<b>1.8. Contribution .....</b>	17
<b>Chapter 2: Unsupervised Machine Learning Based CD-SEM Image Segregator for OPC and Process Window Estimation .....</b>	25
<b>2.1. Introduction .....</b>	26
<b>2.2. Related Work .....</b>	28
<b>2.3. Proposed Method .....</b>	29
<b>2.4. Implementation and Experimental Results .....</b>	31
<b>2.5. Summary .....</b>	40
<b>Chapter 3: SEM Image Denoising With Unsupervised Machine Learning for Better Defect Inspection and Metrology .....</b>	41
<b>3.1. Introduction .....</b>	41
<b>3.2. Related Work .....</b>	44
<b>3.3. Proposed Method .....</b>	45
<b>3.4. Implementation and Experimental Results .....</b>	50
<b>3.5. Summary .....</b>	53
<b>Chapter 4: Deep Learning-Based Defect Classification and Detection in SEM Images .....</b>	54
<b>4.1. Introduction .....</b>	55
<b>4.2. Related Work .....</b>	62

<b>4.3. Proposed Method .....</b>	65
<b>4.3. A. Overview of the RetinaNet Architecture .....</b>	66
<b>4.3. B. Deep Feature Extractor Networks as Backbone .....</b>	67
<b>4.3. C. Mask R-CNN .....</b>	69
<b>4.4. Implementation .....</b>	72
<b>4.4 A. Datasets .....</b>	72
<b>4.4 B. Evaluation Criteria .....</b>	73
<b>4.4 C. Training .....</b>	74
<b>4.5. Experimental Results .....</b>	76
<b>4.5 A. Ensemble_ResNets Defect Detection Performance .....</b>	76
<b>4.5 B. Mask R-CNN Defect Detection Performance .....</b>	84
<b>4.5 C. Deep Learning Denoiser .....</b>	87
<b>4.5 D. Comparison With Conventional Approach/Tools .....</b>	89
<b>4.5 E: Defect Classifier User-Interface (UI) .....</b>	91
<b>4.6. Summary .....</b>	93
<b>Chapter 5: Unsupervised Machine Learning Based SEM Image Denoising for Robust Contour Detection .....</b>	95
<b>5.1. Introduction .....</b>	96
<b>5.2. Related Work .....</b>	98
<b>5.3. Applied Method .....</b>	99
<b>5.4. Implementation and Experimental Results .....</b>	104
<b>5.4 A. Line/Space (L/S) and Contact/Hole (C/H) Layout .....</b>	104
<b>5.4 B. Pillar Layout With Different Scan Types .....</b>	108
<b>5.5. Proposed DL Denoiser Assisted Framework for Robust SEM Contour Extraction .....</b>	115
<b>5.6. Experimental Results .....</b>	116
<b>5.7. Summary .....</b>	118
<b>Chapter 6: Conclusion and Future Work .....</b>	120
<b>Bibliography .....</b>	124
<b>Abstract .....</b>	133
<b>Biographical Sketch .....</b>	135

## List of Tables

<b>Table 2.1.</b> Normalized distance metric for 5 different distance function for randomly selected 10 CD-SEM images .....	35
<b>Table 2.2.</b> Normalized Ground-Truth Mean/CD obtained from conventional Techniques .....	35
<b>Table 4.1.</b> ResNet50, ResNet101, and ResNet152 backbone architecture .....	69
<b>Table 4.2.</b> Defect class labeling convention .....	73
<b>Table 4.3.</b> Data distribution of defect SEM images .....	73
<b>Table 4.4.</b> Implementation results when experimenting with different backbone architectures .....	75
<b>Table 4.5.</b> Test/Validation accuracy of top 3 ResNet architecture backbones .....	75
<b>Table 4.6.</b> Overall test accuracy of proposed RetinaNet [ensemble_ResNet] framework .....	76
<b>Table 4.7.</b> Overall test accuracy of proposed Mask R-CNN based ADCD framework. IOU @ 0.50/AP50 .....	86
<b>Table 5.1.</b> Summary of analysis metric from SEMSuite <sup>TM</sup> (Contact/Hole patterns) .....	107
<b>Table 5.2.</b> Summary of analysis metric from SEMSuite <sup>TM</sup> (Line/Space patterns) .....	108
<b>Table 5.3.</b> Summary of analysis metric from SEMSuite <sup>TM</sup> (Pillar patterns) .....	113
<b>Table 5.4.</b> Summary of analysis metric from our proposed framework in comparison with SEMSuiteTM (Pillar patterns) .....	118

## List of Figures

<b>Figure 1.1.</b> Depiction of Moore's Law and slowdown of its trend due to device scaling innovation challenges .....	1
<b>Figure 1.2.</b> Projection of number of transistors per device (intel products) through innovation following Moore's law .....	2
<b>Figure 1.3.</b> DTCO, STCO and EUV insertion strategy to maintain affordable scaling trend .....	3
<b>Figure 1.4.</b> Conventional lithographic processing steps sequence .....	4
<b>Figure 1.5.</b> Basic lithography equipment .....	5
<b>Figure 1.6.</b> Example of a litho cluster in 300 mm FAB: (a) Coat/Develop Track [SCREEN Sokudo Duo], (b) Litho Exposure tool [ASML TWINSCAN XT:860M] .....	6
<b>Figure 1.7.</b> Illustration of significance of printed patterns to the correct/target size. (a) Target CD, (b) CD-lines too wide, (c) CD-lines too narrow .....	7
<b>Figure 1.8.</b> (a) Top-Down CD-SEM, (b) Cross-section CD-SEM .....	8
<b>Figure 1.9.</b> (a) Best printing image, (b) Image at non-optimal exposure tool settings (off-target CD and some spaces are not open) .....	8
<b>Figure 1.10.</b> Periodic pattern [Line-Space] on Mask. P_min= Pitch (periodicity) .....	9
<b>Figure 1.11.</b> Process-Window example plot for various focus-dose-CD Combinations .....	10
<b>Figure 1.12.</b> Process Latitudes representation .....	10
<b>Figure 1.13.</b> Schematic depiction of quantification of patterned CD and validation of exposure conditions .....	11
<b>Figure 1.14.</b> Schematic of different wavelengths and light sources used in Lithography .....	12
<b>Figure 1.15.</b> Schematic of double patterning strategy .....	13
<b>Figure 1.16.</b> Realistic examples of different litho lenses with varying NA .....	15
<b>Figure 1.17.</b> Schematic illustration between Small NA vs High NA .....	15

<b>Figure 1.18.</b> Schematic illustration for trade-off between NA vs DOF .....	16
<b>Figure 1.19.</b> Schematic illustration for trade-off between NA vs resist thickness .....	17
<b>Figure 1.20.</b> Flow from (a) Circuit design to (b) Mask. Illustration of the steps and models the OPC flow uses to calculate the expected final wafer pattern for a given mask .....	18
<b>Figure 1.21.</b> Flowchart of the proposed Deep learning-based quantification Framework .....	20
<b>Figure 2.1.</b> Optical inspection determined Process Window .....	27
<b>Figure 2.2.</b> Generative Var-AE Model architecture .....	29
<b>Figure 2.3.</b> The Encoder model compresses image data into a latent vector space ( $Z$ ). The Decoder model reconstructs the image ( $\hat{X}$ ) given the learned hidden representations .....	30
<b>Figure 2.4.</b> (a) Training images, (b) Learned/Reconstructed images by the Model .....	32
<b>Figure 2.5.</b> (a) Classification and Recommendation for CD (line-space) image pattern at dose: 42.0, focus: -0.09. The first 10 best recommended similar printable CD-SEM patterns (images) has been shown out of 230 diverse datasets of CD-SEM images. (b) Recommendation graph-plot for “query-pattern” index S02-M0231-01MS.tif .....	33
<b>Figure 2.6.</b> (a) Classification and Recommendation for CD (line-space) image pattern at dose: 42.0, focus: -0.09. The first 10 best recommended similar non-printable CD-SEM patterns(images) has been shown out of 230 diverse datasets of CD-SEM images. (b) Recommendation graph-plot for “query-pattern” index S02-M0360-01MS.tif .....	33
<b>Figure 2.7.</b> Wafer-map representation for Conventional technique metric vs. proposed ML recommendation for (a) a very good qualitative Process-Window (best printable feature CD-SEM image). (b) for a near good qualitative Process-Window (a random near best printable feature CD-SEM image). (c) for a worst qualitative Process-Window (random non-printable feature CD-SEM image) .....	38
<b>Figure 2.8.</b> Qualitative Process-Window analysis for a reference Dose-Column: 41.0 with varying focus .....	39
<b>Figure 2.9.</b> Plot for Normalized Mean/CD-ground truth value against proposed ML-Recommendation score for the selected Dose column: 41.0 mJ/cm <sup>2</sup> against varying focus .....	39
<b>Figure 3.1.</b> Noisy CD-SEM images (a) Pitch 32 and (b) Pitch 90 .....	42

<b>Figure 3.2.</b> Conventional vs Proposed network .....	47
<b>Figure 3.3.</b> Blind-spot masking mechanism. (a) Original noisy SEM image. (b) image patch extracted from (a). (c) image is modified by replacing a target pixel (labelled red) with a randomly selected pixel value (labelled green) and thus creating a blind-spot. This modified image is used as input during model training. (d) target patch corresponding to (c) as original input patch without modification .....	47
<b>Figure 3.4.</b> (a) Single noisy SEM image is given to the network model. (b) Denoised image generated by the network model without altering real signal .....	48
<b>Figure 3.5.</b> (a) Noisy SEM image P32 with Film thickness 30 nm (b) Denoised Image .....	48
<b>Figure 3.6.</b> Cropped ROI for (a) Noisy SEM image P28 with Film thickness 8 nm [Thin Resist] (b) Denoised image .....	49
<b>Figure 3.7.</b> (a) Original noisy SEM image ROI [P90]. (b) Line Scan Plot comparison for both noisy and denoised SEM images ROI .....	49
<b>Figure 3.8.</b> SEM image analysis with Fractilia MetroLER library (a) Noisy image (b)Denoised images .....	50
<b>Figure 3.9.</b> Power Spectral Density (PSD) plot for Line-Width-Roughness (LWR) .....	51
<b>Figure 3.10.</b> Power spectral density (PSD) plot for Line-Edge-Roughness (LER) .....	51
<b>Figure 3.11.</b> Unsupervised GMM based CD/Contour extraction: (a) noisy vs (b) denoised for Thin resist low-ft images .....	52
<b>Figure 3.12.</b> Defectivity analysis and accuracy metric comparison for Denoised image against Original noisy image .....	52
<b>Figure 4.1.</b> The Defect Inspection and Review Space .....	56
<b>Figure 4.2.</b> Typical Defects: (a) Bridge, (b) Line-Collapse, (c) Gap & Prob-Gap, and challenging Defects: (d) Gap & Prob-Gap in presence of contrast change (e) Micro-bridges .....	59
<b>Figure 4.3.</b> Proposed ensemble model-based defect detection framework .....	65
<b>Figure 4.4.</b> RetinaNet defect detector architecture .....	66
<b>Figure 4.5.</b> Deep feature extractor networks as backbone .....	68

<b>Figure 4.6.</b> Mask R-CNN framework for defect instance segmentation .....	70
<b>Figure 4.7.</b> Flowchart of Mask R-CNN based ADCD framework .....	71
<b>Figure 4.8.</b> BRIDGE detection results with confidence score. (a) Single Bridge, (b)Multiple Bridges .....	77
<b>Figure 4.9.</b> LINE-COLLAPSE detection results with confidence score. (a)Single Line-Collapse, (b) Multiple Line-Collapses .....	78
<b>Figure 4.10.</b> GAP/BREAK detection results with confidence score. Multiple Line-breaks .....	79
<b>Figure 4.11.</b> Detection results of more challenging Probable NANO-GAP separately in presence of GAP/BREAK. Model shows robustness in detecting relatively few probable Nano-Gap defects in presence of frequent Gap defectivity .....	79
<b>Figure 4.12. (a) (b).</b> Detection of NANO-GAPs and Probable NANO-GAPs in presence of contrast change scenario. Contrast change does not affect defect detection performance of proposed ML model in comparison to conventional approach .....	80
<b>Figure 4.13.</b> Mixed defect detection results with confidence score. (a)Single Line-Collapse, (b) Single Gap/Break .....	81
<b>Figure 4.14.</b> Detection results of more challenging NANO-BRIDGE/MICRO -BRIDGE defectivity .....	81
<b>Figure 4.15.</b> Detection results of more challenging NANO-BRIDGE/MICRO -BRIDGE defectivity on new TEST dataset. Model demonstrates robustness in detecting variable degrees of pixel-level micro-bridge defectivity .....	82
<b>Figure 4.16.</b> Defect detection on Review-SEM images .....	82
<b>Figure 4.17.</b> Defect detection results with proposed Mask R-CNN approach. (a) Line collapse, (c) Single bridge, (e) Thin bridge, and (g) Multi-line non-horizontal bridge. Output masks for corresponding (b) Line collapse, (d) Single bridge, (f) Thin bridge, and (h) Multi-line non-horizontal bridge defects .....	86
<b>Figure 4.18.</b> Defect class prediction with corresponding output mask .....	87
<b>Figure 4.19. (a)</b> Noisy SEM image [P32] with micro/nano-bridges (b)Denoised image .....	87
<b>Figure 4.20.</b> SEM image analysis with Fractilia MetroLER library (a) Noisy image (b) Denoised image. Detected edges in denoised image are with less spikes or almost without spikes in comparison to noisy	

image .....	88
<b>Figure 4.21.</b> Defect inspection on (a) Noisy SEM image [P32] with micro/nano-bridges, (b) Denoised image. Denoising generally helps to remove FP defects (resist footing may appear as micro-bridge defects in presence of stochastic noise on structure pixels) .....	88
<b>Figure 4.22.</b> Defect detection on same Noisy SEM image [P32] with micro/nano-bridges: (a) Conventional Tool/approach, (b) Proposed ensemble model-based approach .....	90
<b>Figure 4.23.</b> Defect detection on same Denoised SEM image [P32] with micro/nano-bridges: (a) Conventional Tool/approach, (b) Proposed ensemble model-based approach .....	90
<b>Figure 4.24.</b> Web-based defect inspection app .....	92
<b>Figure 4.25.</b> Essential components of web-based defect inspection app Functionality .....	93
<b>Figure 5.1.</b> Noisy CD-SEM images [Orthogonal pillar array (design CD=20nm, pitch= 40nm) with 3 different scanning directions]: (a) Scan_1, (b) Scan_2, (c) Scan_3 .....	97
<b>Figure 5.2.</b> Proposed deep learning denoiser network .....	102
<b>Figure 5.3.</b> Blind-spot masking mechanism. (a) Original noisy SEM image. (b) image patch extracted from (a). (c) Training patch: A randomly selected pixel intensity value (labelled green) replacing target pixel intensity value (labelled red) towards achieving blind-spot masking mechanism. This modified image is used as input patch during model training. (d) Target patch: [corresponding to (c)] raw original input patch without any modification .....	102
<b>Figure 5.4.</b> (a) Noisy SEM image P28 with film thickness 8 nm [Thin Resist] (b) Denoised image (c) Line-Scan plot comparison for both noisy and denoised SEM images ROI. (d) Extracted and aligned SEM contours GDS [cropped ROI] with Fractilia MetroLER library .....	103
<b>Figure 5.5.</b> (a) Noisy SEM image [T2T CD = 26nm], (b) Denoised image .....	104
<b>Figure 5.6.</b> Orthogonal pillar array [Type: Scan_1] (design CD=20nm, pitch= 40nm): (a) Noisy image, (b) Denoised image .....	104
<b>Figure 5.7.</b> L/S patterns at pitch 26 nm: Contour extraction result analysis on Calibre® SEMSuite™. (a) Noisy image (b) Denoised image .....	105
<b>Figure 5.8.</b> C/H patterns at pitch 28 nm: Contour extraction result analysis on Calibre® SEMSuite™. (a) Noisy image (b) Denoised image .....	105

<b>Figure 5.9.</b> Number of partially extracted contours reduced significantly in denoise images than the ones in noise images for Contact/Hole pattern type .....	106
<b>Figure 5.10.</b> Detection of “# of false micro-bridges” reduces significantly in denoise images than the ones in noise images for Line/Space pattern type .....	106
<b>Figure 5.11.</b> Orthogonal pillar array (design CD=20nm, pitch= 40nm) with 3 different scanning directions: Contour extraction result analysis on Calibre® SEMSuite™. Scan_1: (a) Noisy image (b) Denoised image Scan_2: (c) Noisy image (d) Denoised image Scan_3: (e) Noisy image (f) Denoised image .....	109
<b>Figure 5.12.</b> Denoised Orthogonal pillar array images with 3 different scanning directions: Contour extraction results comparison on Calibre® SEMSuite™ for four different experimental blind-spot pixel manipulation mechanisms as: (1) $p1 \rightarrow uniform\_with\_CP$ (2) $p2 \rightarrow normal\_without\_CP$ (3) $p3 \rightarrow normal\_additive$ (4) $p4 \rightarrow normal\_fitte$ Scan_1: (a) $p1$ , (b) $p2$ , (c) $p3$ , (d) $p4$ [Top Row] Scan_2: (e) $p1$ , (f) $p2$ , (g) $p3$ , (h) $p4$ [Middle Row] Scan_3: (i) $p1$ , (j) $p2$ , (k) $p3$ , (l) $p4$ [Bottom Row] .....	110
<b>Figure 5.13.</b> Plot of “# patterns with properly extracted contours” metric for noisy vs. de-noised images with 3 different scan types (a) Scan_1, (b) Scan_2, (c) Scan_3 on the Pillar patterns. The # patterns with fully extracted contours in denoise images are significantly higher than the ones in noise images across all the scanning modes for four different experimental blind-spot pixel manipulation mechanisms [ $p1, p2, p3, p4$ ] .....	113
<b>Figure 5.14.</b> Plot of “# missed patterns” metric for noisy vs. de-noised images with 3 different scan types on the Pillar patterns. The # missed patterns in denoise images [only shown on $p2$ ] are significantly lower than the ones in noise images across all the scanning modes .....	114
<b>Figure 5.15.</b> Plot of “# patterns with properly extracted contours” metric for noisy vs. de-noised images with 3 different scan types on the Pillar patterns. The # patterns with fully extracted contours in denoise images [only shown on $p2$ ] are significantly higher than the ones in noise images across all the scanning modes .....	114
<b>Figure 5.16.</b> Proposed DL Denoiser assisted Framework .....	116

**Figure 5.17.** L/S pattern [P28\_8nm] (ADI) thin resist: (a) Contour extraction performed using conventional image processing technique on noisy raw CD-SEM image, (b) Contour extraction performed using proposed method on the same noisy raw CD-SEM image ..... 117

**Figure 5.18.** SRAM structure [P28] (ADI): (a) Contour extraction performed using conventional image processing technique on noisy raw CD-SEM image, (b) Contour extraction performed using proposed method on the same noisy raw CD-SEM image ..... 117

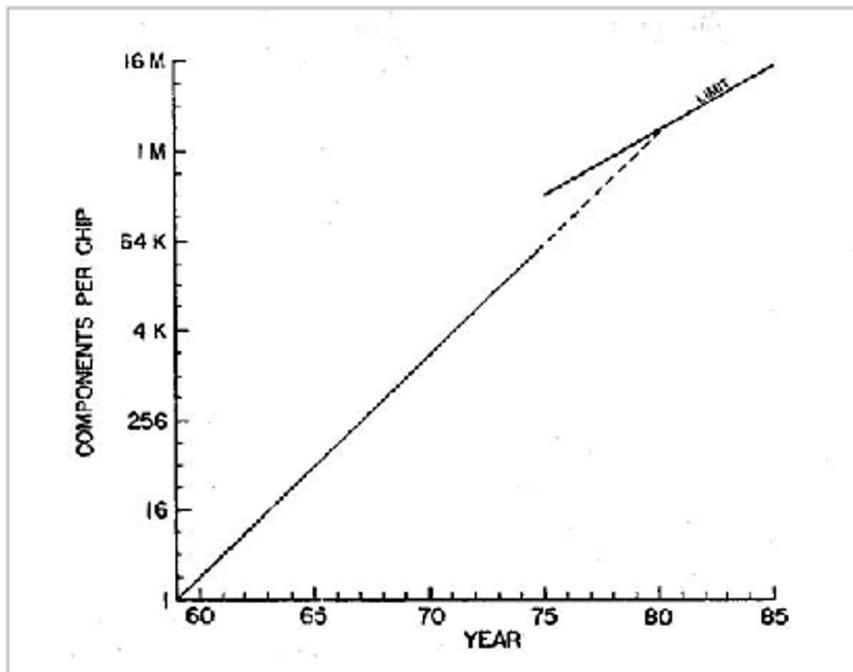
**Figure 5.19.** Orthogonal pillar array (design CD=20nm, pitch= 40nm) with Scan\_1 type: (a) Contour extraction performed using conventional image processing technique on noisy raw CD-SEM image, (b) Contour extraction performed using proposed method on the same noisy raw CD-SEM image ..... 118

## **List of Abbreviations**

ADCD	Automated Defect Classification and Detection
ADI	After Develop Inspection
AE	Auto-Encoder
AEI	After Etch Inspection
AFM	Atomic Force Microscopy
AI	Artificial Intelligence
AP	Average Precision
BEOL	Back-End-Of-Line
CD	Critical Dimension
CNN	Convolutional Neural Network
DOF	Depth of Focus
DL	Deep Learning
EBL	Electron Beam Lithography
EDA	Electronic Design Automation
EL	Exposure Latitude
EUV	Extreme ultraviolet lithography
FEOL	Front-End-Of-Line
FN	False Negative
FP	False Positive
GAN	Generative Adversarial Networks
GDS	Graphic Design System
GMM	Gaussian Mixture Model

GPU	Graphics Processing Unit
HVM	High Volume Manufacturing
IC	Integrated Circuit
LCDU	Local CD Uniformity
LER	Line-Edge-Roughness
LWR	Line-Width-Roughness
ML	Machine Learning
NA	Numerical aperture
OPC	Optical Proximity Correction
OVL	Overlay
PSD	Power Spectral Density
PW	Process Window
RELU	Rectified Linear Unit
SOTA	State-of-the-Art
SNR	Signal-to-Noise Ratio
SWA	Side Wall Angle
SEM	Scanning Electron Microscope
SWA	Side Wall Angle
TEM	Transmission Electron Microscopy
TN	True Negative
TP	True Positive
UI	User-Interface
VAR-AE	Variational Auto-Encoder
2D	Two Dimensional
3D	Three Dimensional

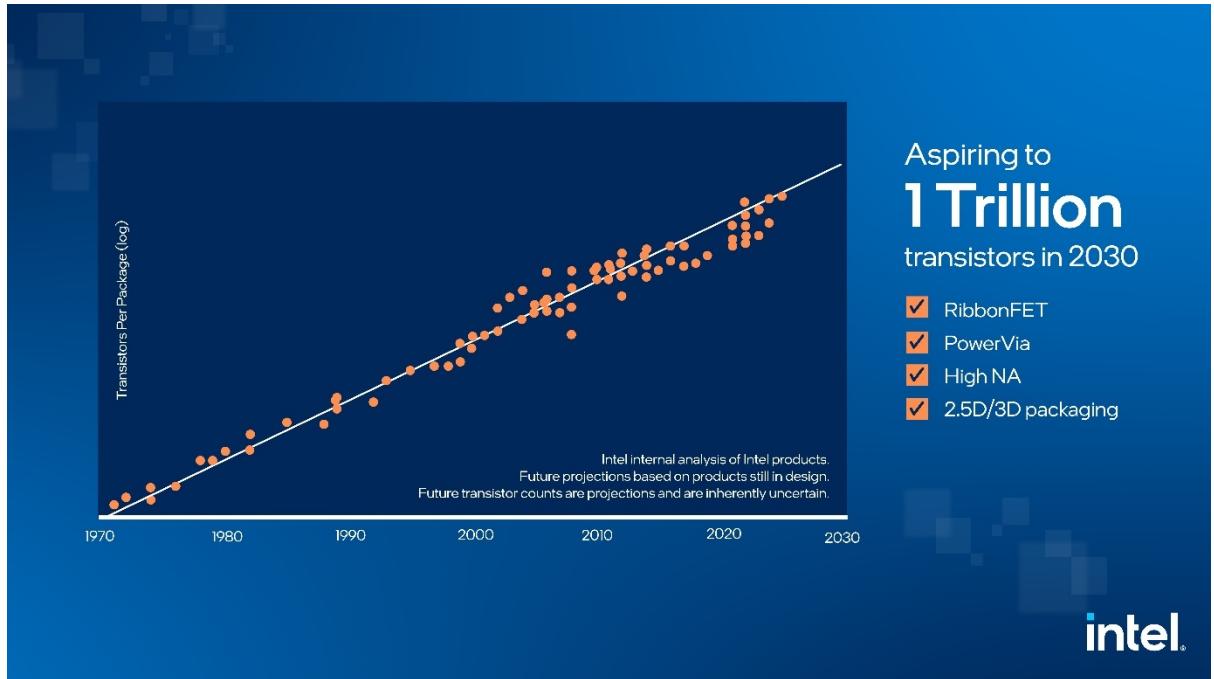
## Chapter 1: Introduction



**Figure 1.1.** Depiction of Moore's Law and slowdown of its trend due to device scaling innovation challenges [1].

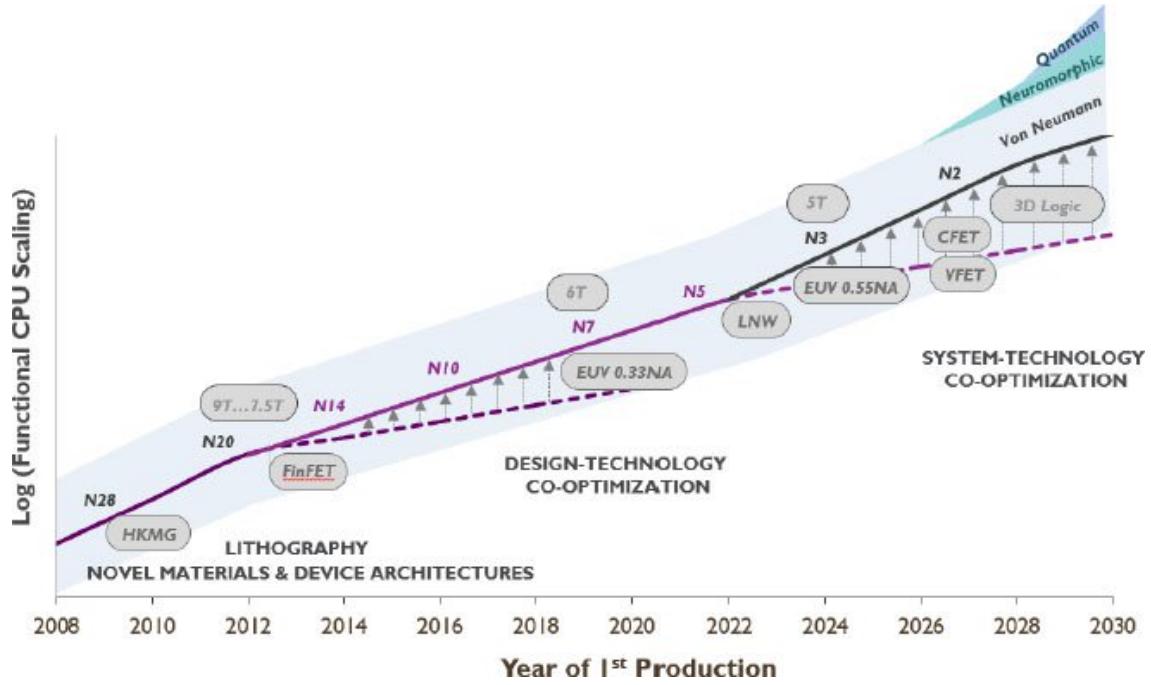
Moore's law, as depicted in Fig. 1.1, is usually referenced to Gordon Moore's prediction [2], that number of transistors and other electronic elements per chip would double approximately every two years. Cramming more transistors in a single integrated circuit (IC) enabled enhanced performance while decreasing cost of it, over the years. Fig. 1.2 depicts projection of number of transistors per device (intel internal analysis for intel products) by 2030. However, this can be achieved through constant shrink from node to node which is made possible with constant pitch scaling (i.e. reduced chip area by 50% per technology node). To understand the impact of technology node transitions, power-performance-area-cost (PPAC) is used as a key metric. However, Moore's law has slowed down over the years due to innovation challenges in dimensional scaling, material search and process technology, respectively. Additionally, (a) increasing costs involved with new technology node transfer, (b) associated financial risks with new wafer fab facility, and

(c) overall challenges of grasping the intricacy of advanced node wafer fabrication have attributed further slowdown of Moore's law.



**Figure 1.2.** Projection of number of transistors per device (intel products) through innovation following Moore's law [3].

Research and innovation, at each critical technical bottleneck interface, along with increasing demand of enormous computing power at low cost, has always pushed Moore's law beyond its limit. To reduce the increased cost and burden of the process enablement, a new paradigm has been proposed as Design-Technology-Co-optimization (DTCO) [4]. This strategy generally driven by patterning technology, which influences the design strategy and cost-of-ownership. Furthermore, System-Technology-Co-optimization (STCO) has been proposed as advanced alternative paradigm to deal with imminent DTCO challenges as shown in **Fig. 1.3**. Moore's law has been realized and driven by multitude of approaches as continuous pitch scaling, new devices, new switches, 2D/3D stacking technologies as well as convergence of technologies as material science, biology, semiconductors, and very recently artificial intelligence.



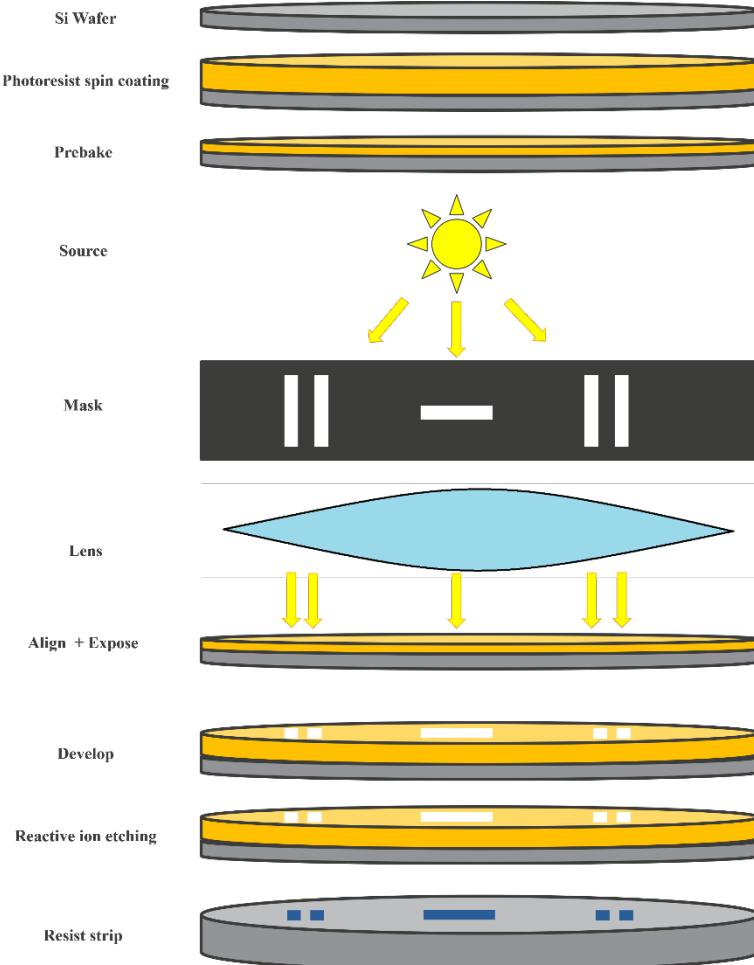
**Figure 1.3.** DTCO, STCO and EUV insertion strategy to maintain affordable scaling trend [4].

This chapter studies the role of lithography in semiconductor manufacturing as well as major concepts and terminologies in the lithography process. It investigates major innovative approaches and challenges specific to lithography that should be considered in making of microchips, and evaluation parameters. In order to solve any or all of these challenges, it is required to know their sources. The chapter starts by presenting conventional lithographic processing steps sequence as well as basic lithography equipment followed by interdisciplinary challenges. Overall, this chapter presents a review of quantification of patterned CD and validation of exposure conditions and finally how our proposed deep learning-based approach increases the throughput to improve the technology and cost concerns associated with the pattern quantification step.

## 1.1. Role of Lithography in Semiconductor Manufacturing

**1.1.1. Optical lithography.** Optical lithography defines a complex process of transferring the geometric patterns of integrated circuits and its peripherals onto a silicon

wafer. Fig. 1.4 demonstrates the conventional lithographic processing steps sequence. The four main elements of this process



**Figure 1.4.** Conventional lithographic processing steps sequence.

Fig. 1.4 demonstrates the conventional lithographic processing steps sequence. The four main elements of this process are: (a) Si wafers/substrates, (b) a selective photoresist (positive or negative), (c) photomask(s) and (d) lens, respectively. The number of required masks or a mask-set depends on the number of required layers for a complete design. The pattern formation steps on a wafer can be summarized as follows:

1. **Wafer coating:** wafer is coated with a photosensitive layer, called a photoresist.

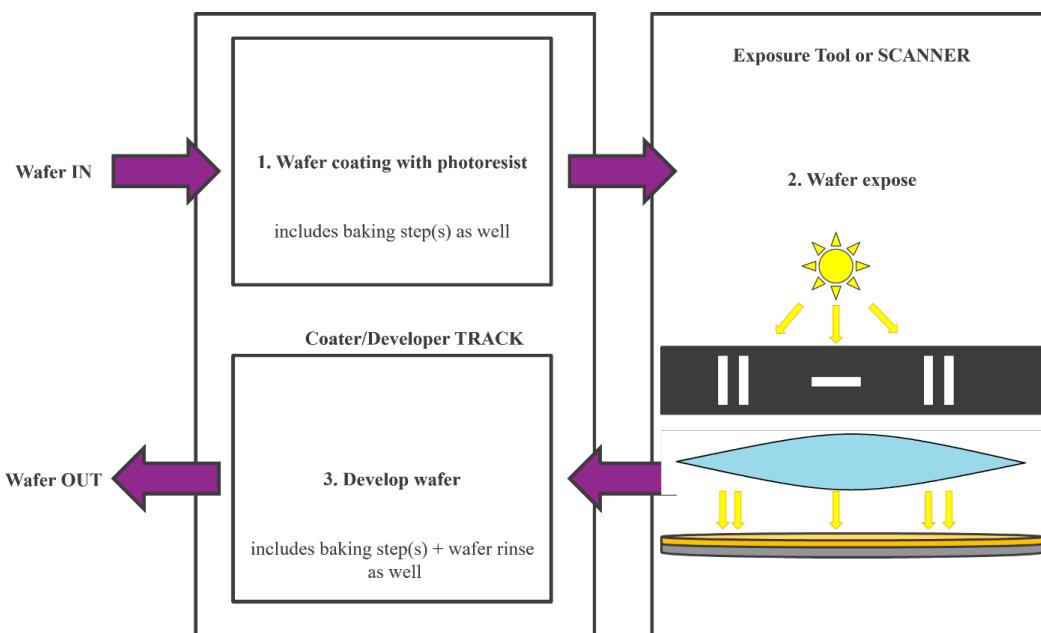
**2. Wafer exposure:** an optical image of the mask pattern is made on the wafer as exposed resist is chemically changed.

**3. Wafer development:** Through wet development as the exposed resist is removed.

**4. Etch:** The goal is to etch into layer under the resist, usually by ion-bombardment in a plasma.

**5. Remove resist:** Once the desired pattern has been transferred in under layer, the remaining resist is stripped.

**1.1.2. Basic lithography equipment.** Fig. 1.5 shows an overview of basic lithography equipment. The sequence of steps or flow we described above, to make microchips, basically

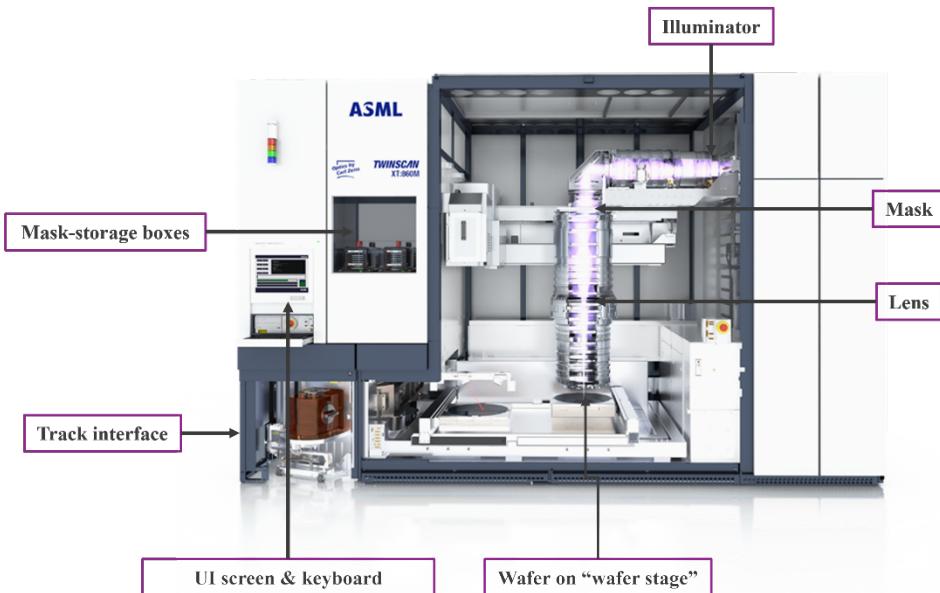


**Figure 1.5.** Basic lithography equipment.

makes use of two different tools broadly as: (a) coater/developer Track, and (b) exposure tool or Scanner. Everything that is connected to the resists and chemistry and making that work is happening in the Track, whereas, exposing itself is happening in the exposure tool or Scanner. Fig. 1.6 (a) (b) shows example of a litho cluster in 300 mm fab.



(a)



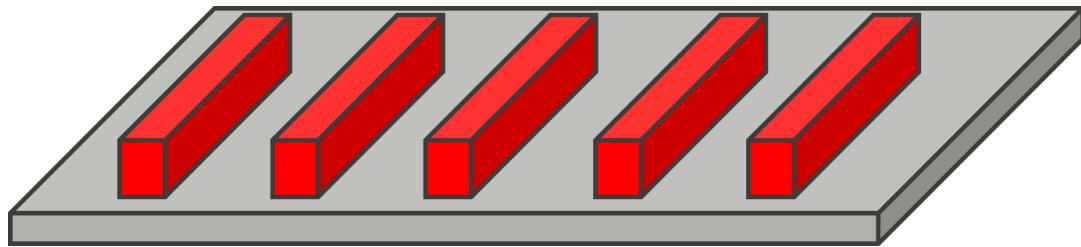
(b)

**Figure 1.6.** Example of a litho cluster in 300 mm FAB: (a) Coat/Develop Track [SCREEN Sokudo Duo] [5], (b) Litho Exposure tool [ASML TWINSCAN XT:860M] [6].

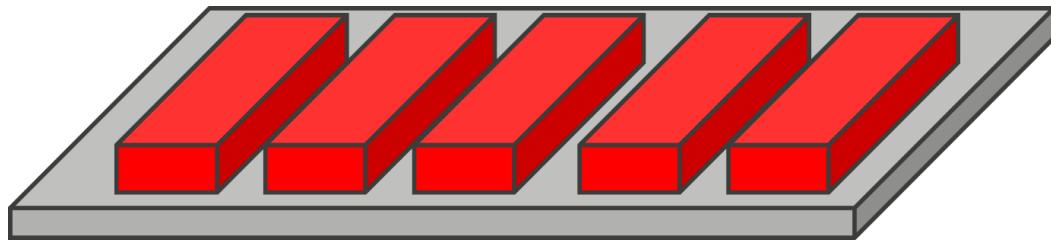
## 1.2. Critical Dimension (CD): Size of the Structures we Print

Following the lithographic processing steps sequence with litho-cluster tools in a fab does not guarantee, always and automatically print to the correct dimension. Therefore,

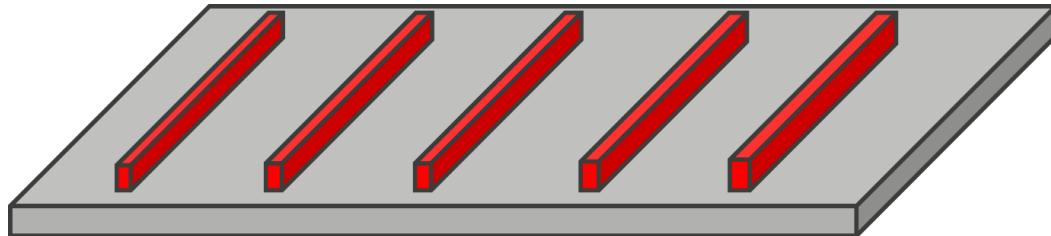
if the geometrical structures are not patterned to the intended size, the electrical performance of the device will be affected. Considering the worst-case scenario, it is possible that the device will not work at all following a missing electrical connection or an electrical short.



(a)



(b)

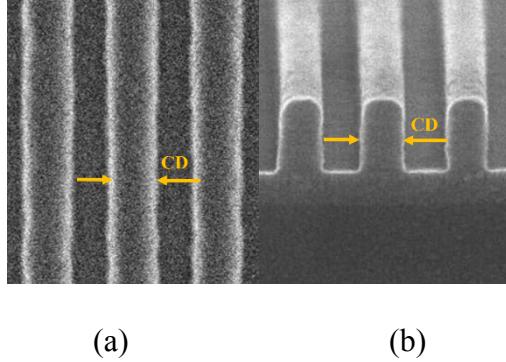


(c)

**Figure 1.7.** Illustration of significance of printed patterns to the correct/target size. (a) Target CD, (b) CD-lines too wide, (c) CD-lines too narrow.

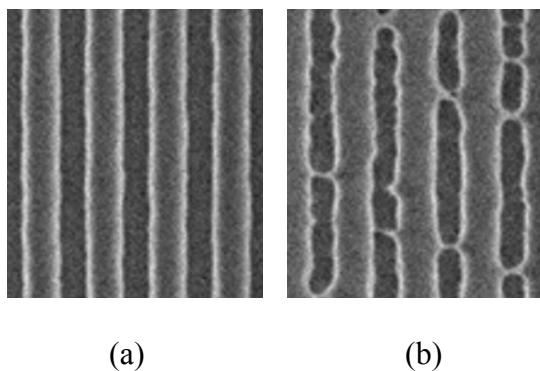
**Fig. 1.7 (b)(c)** demonstrate the importance of patterning metal lines (red lines) for making electrical connections to the correct size **Fig. 1.7 (a)**. **Fig. 1.7 (b)** depicts a scenario, where the lines are too wide and therefore enabling the risk for electrical shorts between multiple lines. Fig. 1.7 (c) shows another consequence, if the lines are too narrow and therefore the connections are of high resistance. Therefore, once the litho steps are done (each iterative cycle), we are required to measure the printed dimensions on the wafer/die. There are two

possible ways for measuring these dimensions, using a Secondary Electron Microscope (SEM) tool as: (a) top-down as shown in Fig. 1.8 (a), and (b) cross-section as shown in Fig. 1.8 (b). The later option is always destructive, and we mostly avoid it.



**Figure 1.8.** (a) Top-Down CD-SEM [56], (b) Cross-section CD-SEM [7].

In general, while measuring the CD, we have a target-CD for every critical structure, and we only accept a certain deviation [ $\pm 10\%$ ] on the target-CD, termed as CD-tolerance. After exposing every lot in Litho, a limited number of structures are scrutinized in the top-down CD-SEM to ensure that measured CD follows specified CD-tolerance. Two significant reasons that patterned CD cannot always be right as: (1) patterned images created by the optical lenses on the wafer cannot be perfect copies of the mask patterns, and (2) if exposure tool settings are not optimal, which will lead further problematic scenarios as dose/focus errors, image quality will also be affected as shown in Fig. 1.9 (a) (b).

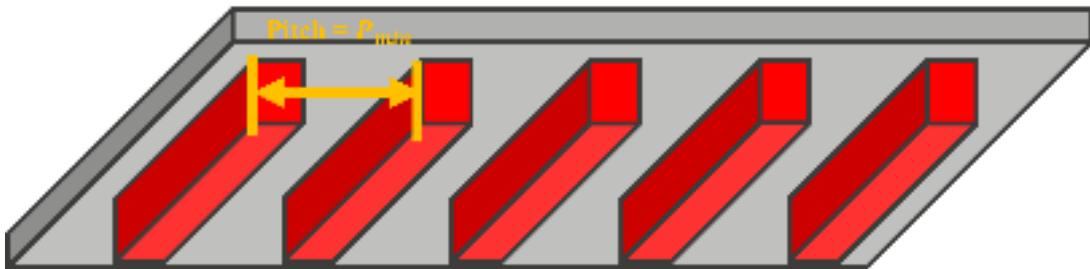


**Figure 1.9.** (a) Best printing image, (b) Image at non-optimal exposure tool settings (off-target CD and some spaces are not open)

### 1.3. Rayleigh Equation

As discussed above, in the context that patterned images created by the optical lenses on the wafer cannot be perfect copies of the mask patterns as there is a physical limit to the “smallest pattern” that is allowed to be printed with a given lens. This limit is driven by a simple equation, known as Rayleigh equation [8], which offers the smallest printable pitch/resolution limit as:

$$\frac{P_{min}}{2} = 0.25 \frac{\lambda}{NA} \quad \text{Eq. 1.1}$$

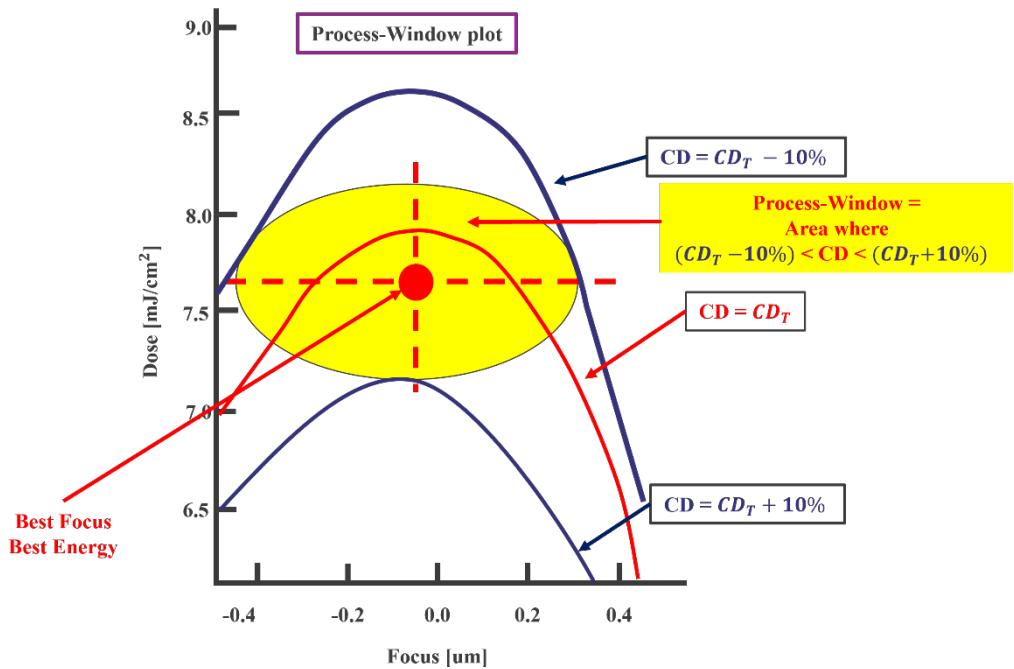


**Figure 1.10.** Periodic pattern [Line-Space] on Mask.  $P_{min}$ = Pitch (periodicity).

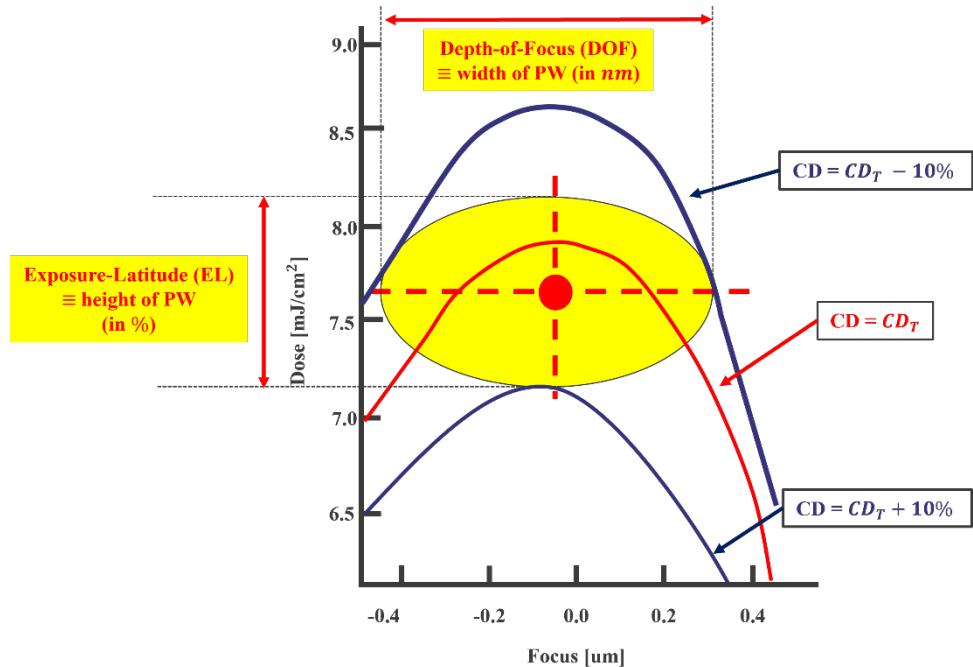
Fig. 1.10 demonstrates the periodic pattern on mask for a given  $\lambda$ , wavelength used for the exposure and  $NA$ , numerical aperture of the lens. The numerical aperture also quantifies the size of the lens. To enforce patterning of small pitches, we generally have two alternatives as: (a) adopt a small exposure wavelength  $\lambda$ , and/or (b) high NA.

### 1.4. Process Window

A process window, a graphical plot in Dose-Focus space, establishes the region for which Focus-Dose combinations the measured CD equals the target CD, as well as the CD-tolerance [ $\pm 10\%$ ]. We can find the “best focus” and “best dose” at the center point of this defined process window as shown in Fig. 1.11.



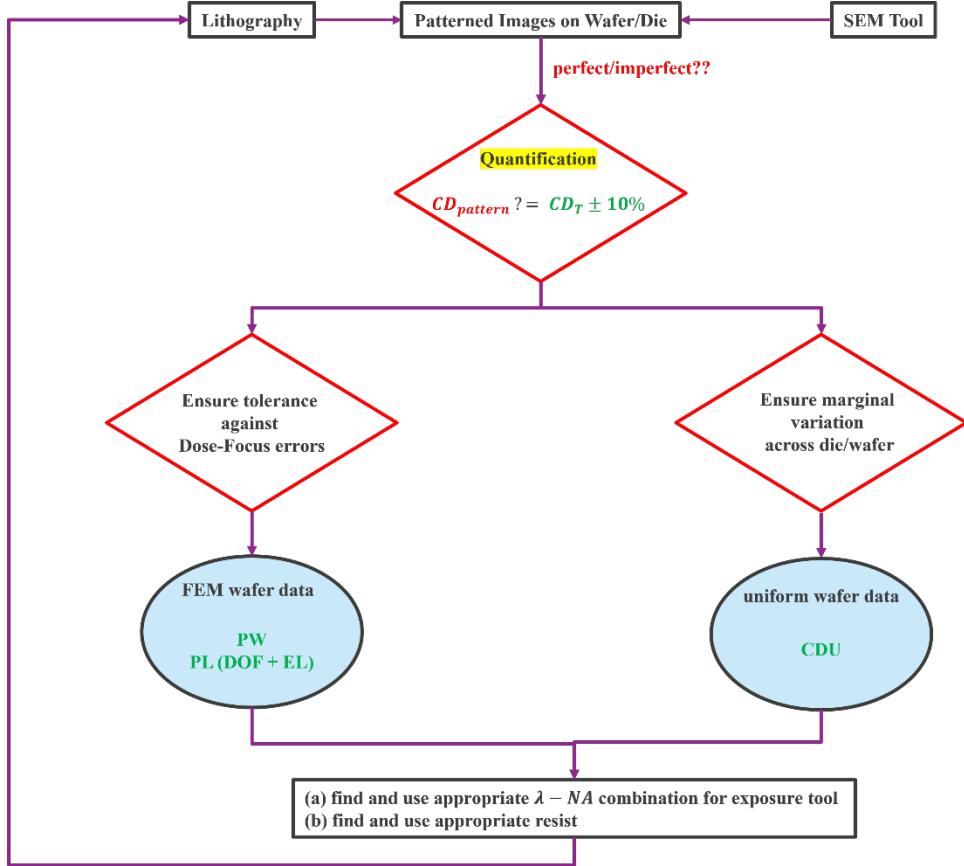
**Figure 1.11.** Process-Window example plot for various focus-dose-CD combinations.



**Figure 1.12.** Process Latitudes representation.

Process latitudes provide us the width and height of the process window. The width of the process window represents the depth-of-focus (DOF), and height represents the exposure-latitude (EL) as shown in Fig. 1.12. These parameters enable the engineers/researchers to find appropriate litho materials (such as resist) as well as tolerance around best dose-focus.

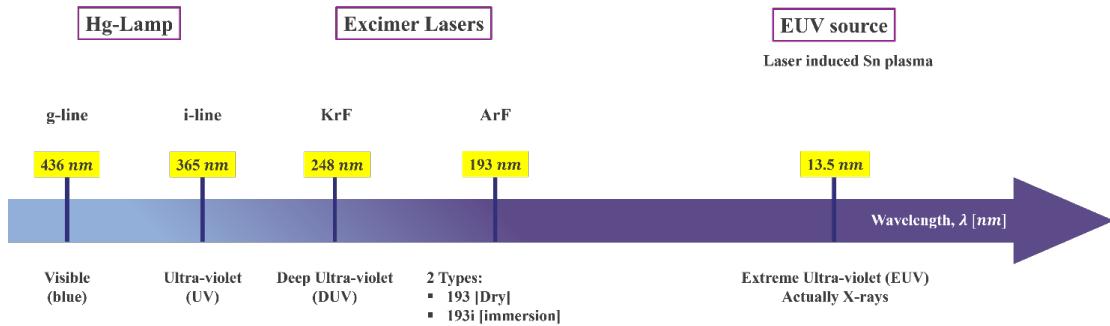
This tolerance limit aids in challenging scenario as: (a) real dose-focus values always deviates from the ideal/theoretical values, and (b) focus and dose always vary at different die/wafer positions as well as between the wafers of the same batch.



**Figure 1.13.** Schematic depiction of quantification of patterned CD and validation of exposure conditions.

Fig. 1.13 demonstrates the summary of the above discussed section as well as represents the quantification procedure for patterned CD against a target-CD and validation of the exposure conditions of litho-cluster tools.

## 1.5. Wavelengths and Light Sources used in Lithography

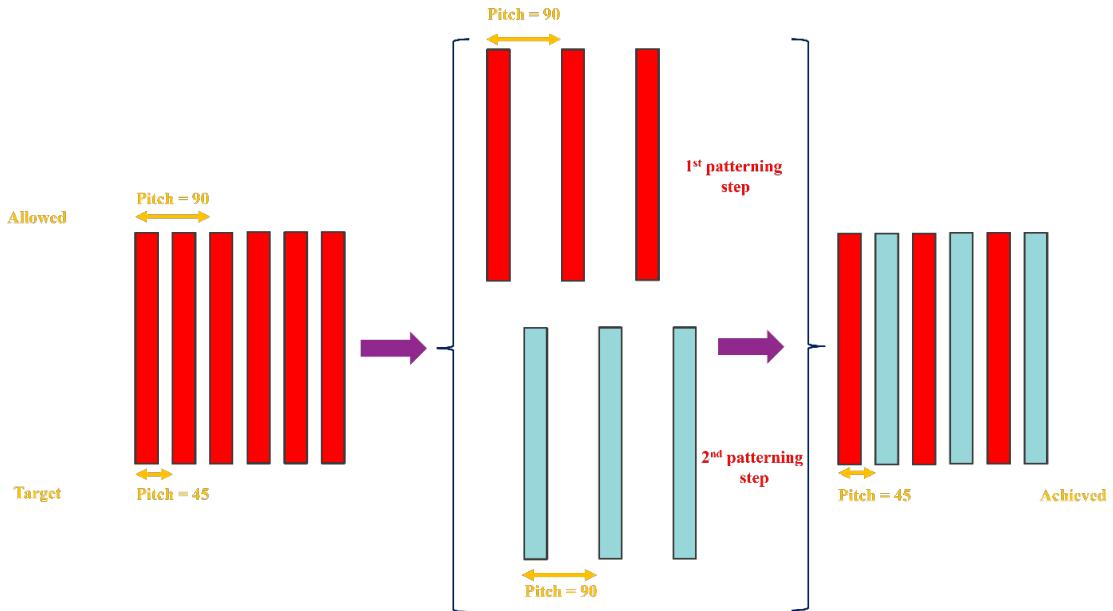


**Figure 1.14.** Schematic of different wavelengths and light sources used in lithography.

Fig. 1.14 shows different wavelengths and light sources used in lithography. We have explicitly segmented three different lithography paradigms as: (a) Hg-lamp based, (b) Excimer laser based and (c) EUV source based, respectively. The light source of an i-line system is a mercury (Hg) lamp. However, Hg lamp emits several wavelengths, but i-line tool filters out all other wavelengths except the light around  $365\text{ nm}$ . The only difference between i-line and g-line is that, in g-line we used previously the peak around  $436\text{ nm}$  wavelength. g-line is no longer used in lithography at present days. For excimer laser, previous Hg lamp source is replaced with a chamber filled with a mix of gases containing Kr and  $\text{F}_2$  and/or Ar and  $\text{F}_2$ , respectively. By generating an electrical discharge, short-lived molecules are created as KrF ( $\text{Kr} + \text{F}_2$ ) or ArF ( $\text{Ar} + \text{F}_2$ ), which emit two specified laser-wavelengths as  $248\text{ nm}$  (for KrF) and  $193\text{ nm}$  (for ArF), respectively. Excimer lasers are “pulsed lasers” by nature as they do not emit light continuously, but rather in short bursts of length  $\approx 30\text{ ns}$  long. Against prior two methods, the wavelength generation process is different as well as complex in EUV. First, a droplet generator releases a stream of small-scale Tin (Sn) droplets. Secondly, a high-power laser pulse hits each droplet to create a “plasma”, which emits the desired wavelength of  $13.5\text{ nm}$ . Finally, a collector mirror accumulates as much as possible of this light and directs it into the illuminator of the scanner.

## 1.6. Double Patterning

Advancing the resolution limit further to print smaller structures can be made possible by (a) building lenses with ever larger  $NA$ , and/or (b) adopting a shorter wavelength  $\lambda$ . At some critical technical bottleneck, it is always not possible to increase  $NA$  beyond its present limit as well as unattainability of shorter wavelengths. If increasing wavelength  $\lambda$  and  $NA$  of lens no longer possible, double patterning (DP) can be considered as another way to increase the resolution limit. Double patterning enables to print too dense patterns, which are not possible to print directly due to resolution limit, by splitting in two or multiple less dense patterns and then printing them separately.



**Figure 1.15.** Schematic of double patterning strategy.

Fig. 1.15 demonstrates the double patterning strategy. For an example, we want to print a line-space pattern with  $45\text{ nm}$  pitch with  $193i\text{ nm}$  tool, considering EUV tool does not exist. This can be considered as a critical technical bottleneck, as with  $193i\text{ nm}$  tool, the allowed resolution limit is  $\approx 80\text{ nm}$ . As demonstrated in the Fig. 1.15, we resolved the problem by patterning the same layer two times, each time printing  $90\text{ nm}$  line-space pitch. As a result, we achieved desired  $45\text{ nm}$  pitch by combining two patterns together, beyond  $193i\text{ nm}$  tool resolution limit. There are multiple scientific methodologies [9] to print

complicated patterns by adopting the double patterning strategy as: (1) Litho-Etch-Litho-Etch (LELE), two separate litho steps, each time followed by an etch step; (2) Self-Aligned-Double-Patterning (SADP), which needs only a single litho step; and (3) Self-Aligned-Quadruple-Patterning (SAQP). The technical details of these approaches are beyond the scope of this thesis work. The key disadvantages are as follows:

- (1) Double patterning requires more complicated wafer stacks (including hard masks and/or etch-stop layers) than single patterning.
- (2) Final pattern is sensitive to alignment errors between two or multiple layers.

### **1.7. Importance of High NA and Practical Challenges**

For any given lens, certain structures (corresponding pitches) cannot be printed at all, as governed by Rayleigh Equation in sec 1.3:

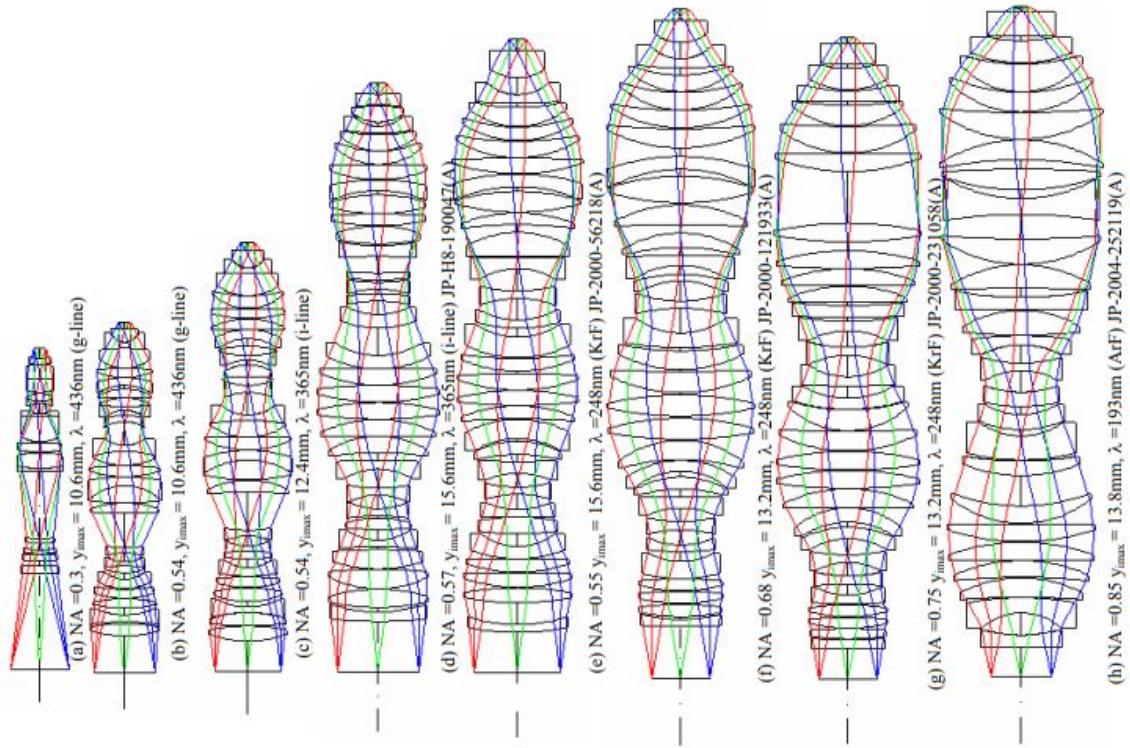
$$\text{Resolution limit} \propto \frac{\lambda}{NA} \quad \text{Eq. 1.2}$$

Therefore, printing small structures requires a high  $NA$ , given a fixed wavelength  $\lambda$ . Fig. 1.16 shows the realistic examples of different litho lenses. As well as we can see that with varying  $NA$  (mostly increasing  $NA$ ), both the size and design complexity of these litho lenses are increasing.

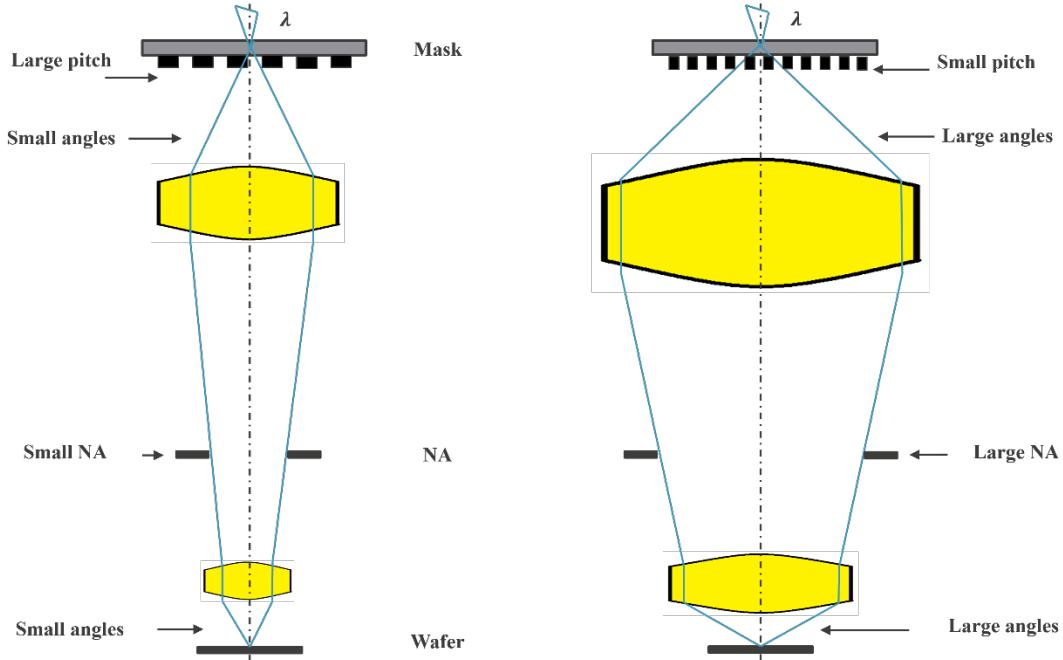
A more practical way of rewriting Eq. 1.1 is:

$$\frac{P_{min}}{2} = k_1 \frac{\lambda}{NA} \quad \text{Eq. 1.3}$$

where,  $k_1$  may range from 0.3 to 0.4. Fig. 1.17 shows key differences between small  $NA$  and high  $NA$  configurations.

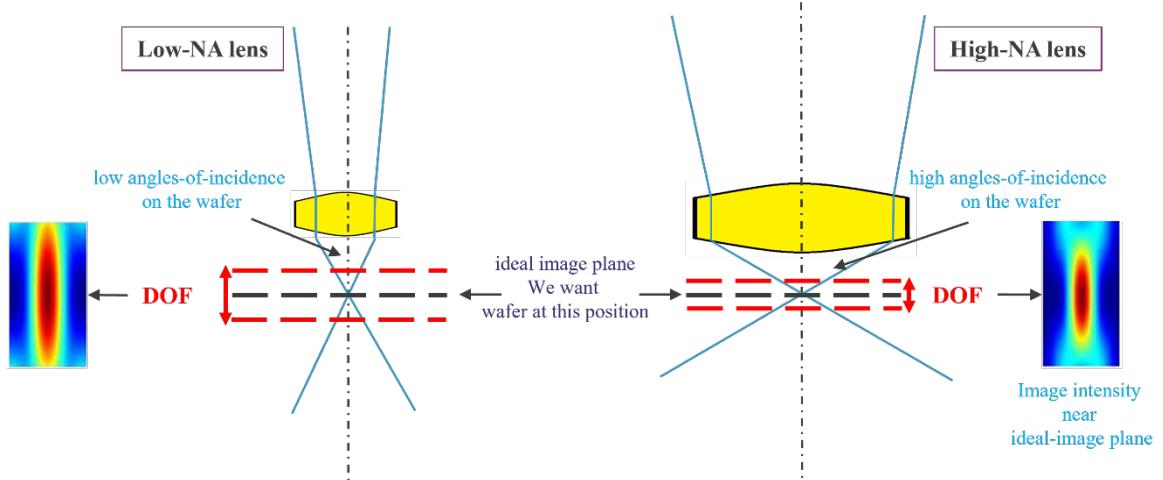


**Figure 1.16.** Realistic examples of different litho lenses with varying  $NA$ . Reprinted from [10].



**Figure 1.17.** Schematic illustration between Small  $NA$  vs High  $NA$ .

**1.7.1. High NA and depth-of-focus.** Fig. 1.18 shows trade-off between choosing a  $NA$  settings (low, high) and DOF. The correlation between DOF and  $NA$  can be represented as:



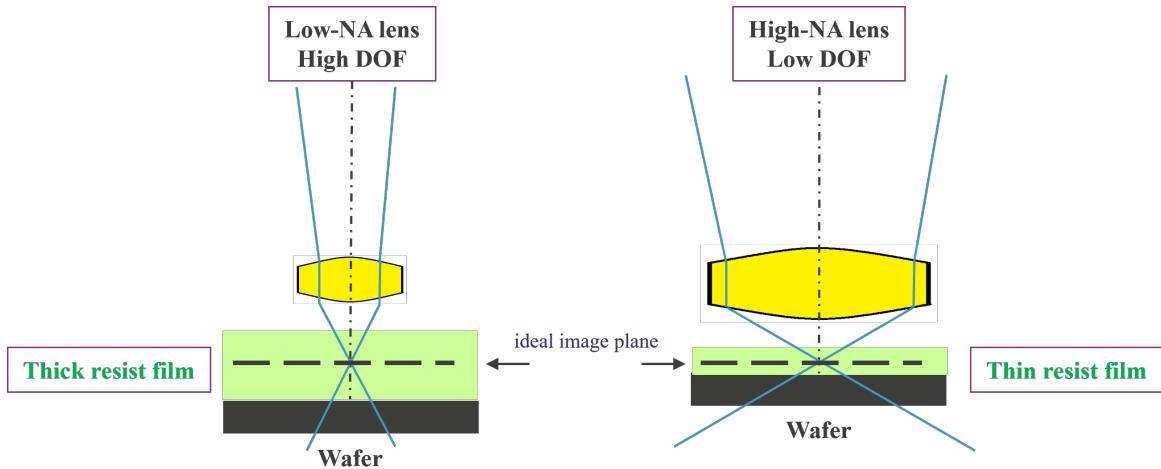
**Figure 1.18.** Schematic illustration for trade-off between  $NA$  vs DOF.

$$DOF \propto \eta \frac{\lambda}{NA^2} \quad \text{Eq. 1.4}$$

for dry lithography, we choose  $\eta = 1.0$  and for immersion lithography,  $\eta = 1.44$ , respectively.

As we can see from Fig. 1.18, if due to residual error wafers cannot be placed at the ideal plane (which can be considered as an obvious scenario), and wafer is a little bit out-of-focus (vertically shifted up/down from the ideal plane), we cannot expect the images as good as the ideal plane (sharpest image). However, low  $NA$  system provides relatively larger margin (i.e., beam diverges less fast if we move away vertically from the ideal plane) than high  $NA$  system. Therefore, low  $NA$  system has larger DOF than high  $NA$  system, which means for the later the images degrade much faster than the former. Therefore, choosing the  $NA$  metric demands a careful investigation between resolution and DOF.

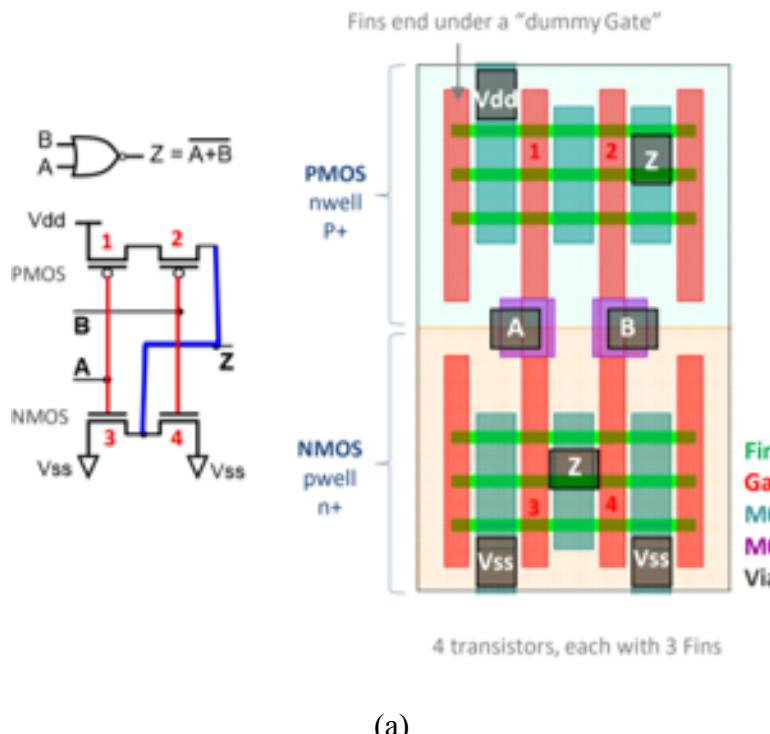
**1.7.2. High NA and resist thickness.** The previous relationship between DOF and



**Figure 1.19.** Schematic illustration for trade-off between  $NA$  vs resist thickness.

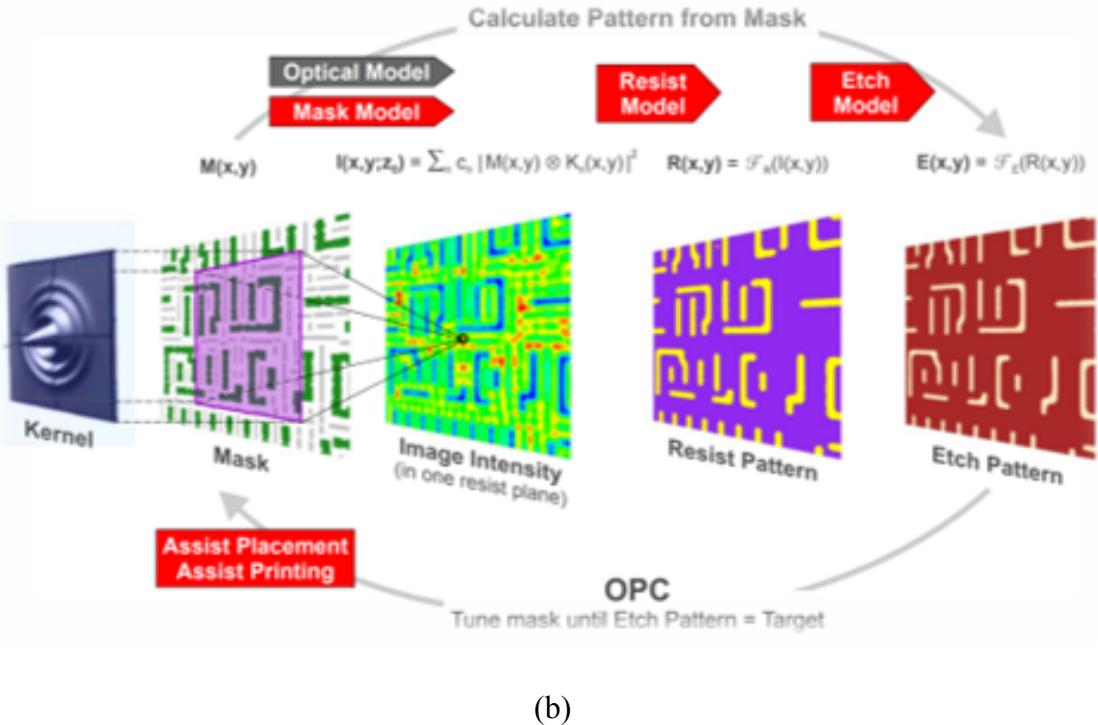
$NA$ , represented in Eq. 1.4 holds true for resist thickness as well. Therefore, low  $NA$  systems work with thicker resist, whereas, for high  $NA$  system, we must use thin resist as shown in Fig. 1.19.

## 1.8. Contribution



4 transistors, each with 3 Fins

(a)



(b)

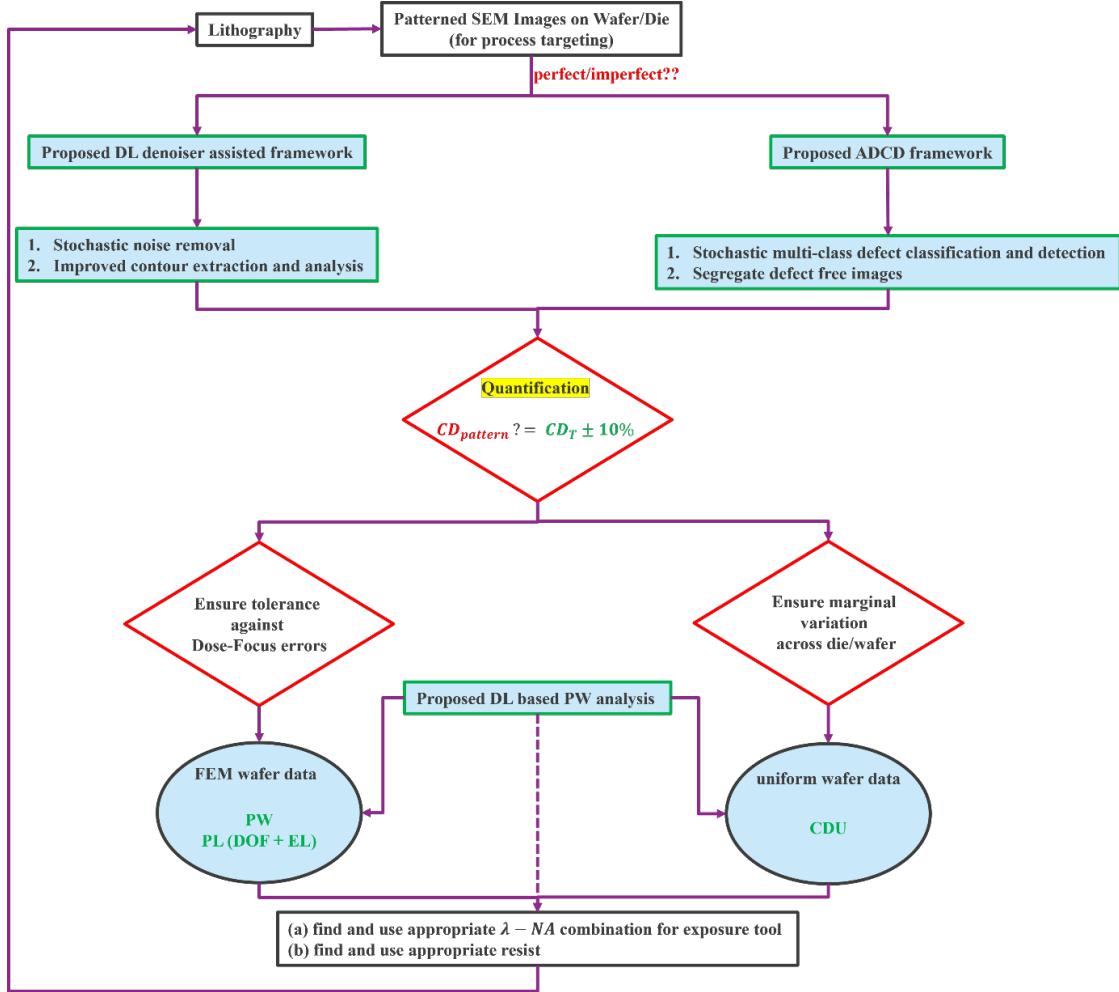
**Figure 1.20.** Flow from (a) Circuit design to (b) Mask. Illustration of the steps and models the OPC flow uses to calculate the expected final wafer pattern for a given mask [11].

From circuit design towards making a mask is a complicated flow. Design needs to be decomposed to different shapes before we can print them. A mask is used to print a particular design feature. Before we can get our pattern, number of iterations required between design and mask making, which involves metrology almost in every step. Fig 1.20 illustrates the involved steps and models the OPC flow uses to calculate the expected final wafer pattern for a given mask. Except the optical model, all models need to be calibrated on measured wafer data. Not all models shown in this scheme are mandatory, however, but using them improves the accuracy and robustness of the final solution. The only wafer data that we get at this stage is extensive SEM based measurements. Metrology and/or inspection for these printed patterns is key to have good models and good processes, which finally results in good yield. However, with continuous node shrink, metrology/inspection are becoming more challenging with proposed high-NA EUV. One of the key requirements, for printing smaller features with High NA scanners, is thinner resist, as discussed in

section 1.7.2. This requirement generally leads to image contrast loss, which in turn stimulates errors in metrology/defect inspection.

In this research work, our motivation was to investigate the challenges of e-beam metrology as well as complex requirements of high NA EUV lithography, and to propose improved solutions of e-beam inspection and metrology problems by applying deep learning-based algorithms and architectures in semiconductor industry. The flow chart of our proposed work is shown in Fig. 1.21. Against the conventional quantification framework as shown in Fig. 1.13, our proposed framework includes deep learning denoiser as assist tool for (a) improved defect inspection, and (b) improved contour detection/extraction. The significant difference also lies in the fact that in conventional framework, SEM tool is giving additional inputs (measurement data) along with patterned SEM images, whereas our proposed framework only receives patterned SEM images for litho process targeting. At first, we denoise the SEM images (ADI) to optimize the significance of stochastic noise on structured pixels, without altering the actual morphology of the patterns. The ADCD method checks different stochastic defects. If there is an identified defect, it separates that defect image from the rest. This step finally segregates and save the patterned SEM images in different folders according to their corresponding defect categories as well as defect free images in a separate folder. The framework includes improved contour extraction algorithm to extract contours from the noisy patterned SEM images itself. The contour extraction approach has some interesting properties as the absence of external user input or metadata (like GDSII / OASIS data, CSV-like meta-data set etc.) to extract and analyse information from noisy SEM images. The second is the absence of SEM tool measurement data. Finally, autoencoder based approach is proposed to automatically define defect process window replacing manual measurement from FEM wafer. Therefore, our proposed deep learning-

based quantification framework can be applied to extract repeatable and accurate CDSEM metrology information for high NA EUV (with low contrast and noisy images).



**Figure 1.21.** Flowchart of the proposed Deep learning-based quantification framework.

The main contributions of this work are summarized as follows:

- Proposing autoencoder based machine learning algorithm for pre-processing OPC data, primarily for data clean-up. Extending the method to automatically define defect process windows (TPs, TNs and most importantly FPs), where TPs are the printable features and TNs and FPs are non-printable or erroneous features, respectively.

- Proposing a novel deep learning denoiser as an assist tool for defect inspection. The proposed method requires single noisy acquisitions to train denoising CNNs without any clean or noiseless ground-truth or synthetic images. The proposed denoising approach reduces only the noise level, without altering the actual morphology of the pattern feature. Denoised images, generated by our proposed method, allows improved identification of stochastic defects as well as to work with thin resists.
- Designing a novel robust supervised deep learning training scheme based on RetinaNet architecture to accurately classify as well as localize different defect types in SEM images in aggressive pitches. We have proposed a preference-based ensemble strategy to combine the output predictions from different experimental backbone architectures towards improved classification and detection accuracy. We have also investigated and demonstrated how the condition influences defect detection scenarios if the image is noisy or denoised and how denoised SEM images are aiding for better metrology and enhanced defect inspection. Finally, we proposed a Mask R-CNN based approach for improved defect instance segmentation in SEM images with precise extent of defect as well as generating a binary mask for each defect category/instance.
- Proposing a novel deep learning denoiser assisted framework for robust SEM contour extraction and analysis for advanced semiconductor node. Our proposed approach facilitates improved contour detection/extraction capability to further provide more accurate metrology data (as estimation of CD/Pitch/resist sidewall/OVL/LER/LWR etc.) for OPC calibration as well as allows identification of defects with improved certainty, while reducing the dependency of SEM image acquisition settings. Our proposed improved contour extraction algorithm enables

to extract contours on the body of noisy raw images itself for categorically different geometrical patterns with a posteriori knowledge derived from its denoised twins.

The dissertation chapters are the following:

- Chapter 2 presents an implementation of unsupervised machine learning based novel approach to automatically analyse the process window. The autoencoder based defect classification shows pattern deviation from nominal, and therefore, can be used to flag outliers and defect free process window. FEM wafer reconstruction validates the method's capability to distinguish small variations. The key idea to define a true process window demands a detailed analysis of critical dimension and other underlying features. The novelty of the proposed approach lies in the fact that (1) with a defect inspection tool this task seems tedious and time consuming and often require human intervention to analyse a large number of features, (2) a CD-SEM based process window analysis might not always match with a defect inspection-based process window. Our generalized auto-encoder based approach does this automatically.
- Chapter 3 presents an implementation of unsupervised deep learning training scheme to denoise SEM images without requiring noiseless ground-truth images in true sense. The proposed approach, therefore, reduced the risk of information loss as well as additive digital artefacts in denoised images, involved with previous cited methodologies. An unsupervised Gaussian Mixture Model based feature edge detection had been proposed to overcome the limitations of conventional edge-detection algorithms like Canny-edge detector in terms of parameter settings. The proposed approach is trainable using a single or few noisy SEM images against

other data greedy approaches. We have validated that the actual morphology of the pattern features in denoised images is unaltered.

- Chapter 4 presents a novel ensemble deep-learning model to solve challenging defect detection problems in SEM images in aggressive pitches. We have trained RetinaNet model using different ResNets, VGGNet, SSD\_MobileNet\_v1 architectures as backbones and proposed a preference-based ensemble strategy to combine the output predictions from different models and achieve better performance on classification and detection of different defects. We have also investigated and demonstrated how the condition influences defect detection scenarios if the image is noisy or denoised and how denoised SEM images are aiding for better metrology and enhanced defect inspection. Finally, we have extended this research further towards improved defect instance segmentation task along with previous classification and detection strategy by implementing Mask-RCNN based approach. By generating a precise extent of binary mask for each defect category/instance, our proposed method enabled to extract and calibrate each segmented mask and quantify the pixels that make up each mask, which in turn enabled us to count each categorical defect instances as well as to calculate the surface area in terms of pixels. These extracted parameters as area/length/width/defect coordinates are effective for potential analysis of root cause of these defects.
- Chapter 5 introduces a novel deep learning denoiser assisted framework for robust SEM contour extraction and analysis for advanced semiconductor node. Firstly, the research demonstrated how the applied deep learning based denoising algorithm is helping to improve the contour detection/extraction capability/accuracy with after development (ADI) SEM images. We have analysed, compared, and validated our

contour extraction results for each noisy/denoised image pair for categorically different geometrical patterns using SEMSuite™ programmable tool. The comparative analysis demonstrated that number of missed patterns and number of false bridges are significantly reduced with denoised images. Most significantly, the proposed approach enables removing stochastic noise which in turn optimizing the varying non-uniform background intensity for better contour detection.

Secondly, this chapter presents an improved contour extraction algorithm capable to extract contours on the body of noisy raw image itself with a posteriori knowledge derived from its denoised twins. The proposed method is independent of external user input or metadata (design GDS) to extract and analyse contour information from noisy SEM images, with accuracy in close proximity with design data.

## **Chapter 2: Unsupervised Machine Learning Based CD-SEM Image Segregator for OPC and Process Window Estimation**

As we are stepping towards sub-10 nm nodes, process window monitoring for systematic defects is becoming more and more critical. In traditional process window excursion and control (PWEC) methods often optical defect inspection is done on a focus and dose modulated wafer first. Once the different systematic defects are detected in a particular focus/energy die, we flag the repeating defect locations as potential hotspots and rank them based on how early/late they fail in a focus/energy modulated column. So, during this first pass we get a rough idea of which locations are failing. However, due to limited resolution of optical tools, the true process window can only be gathered during a second pass with an ebeam tool. The key idea to define a true process window demands a detailed analysis of CD and other underlying features. We have proposed a new method of analyzing the process window with an unsupervised machine learning approach. Our proposed algorithm will extract the underlying key features and encode these to latent feature vectors or latent vector space instead of the conventional CD, given a dataset of thousands of CD-SEM images, and then rank the images based on a similarity index and then to automatically determine the process window. This work addresses the following problems (1) with a defect inspection tool this task seems tedious and time consuming and often require human intervention to analyze a large number of features, (2) a CD-SEM based process window analysis might not always match with a defect inspection-based process window. Our generalized variational auto-encoder based approach does this automatically. Also, we have analyzed and validated our result against conventional approach.

---

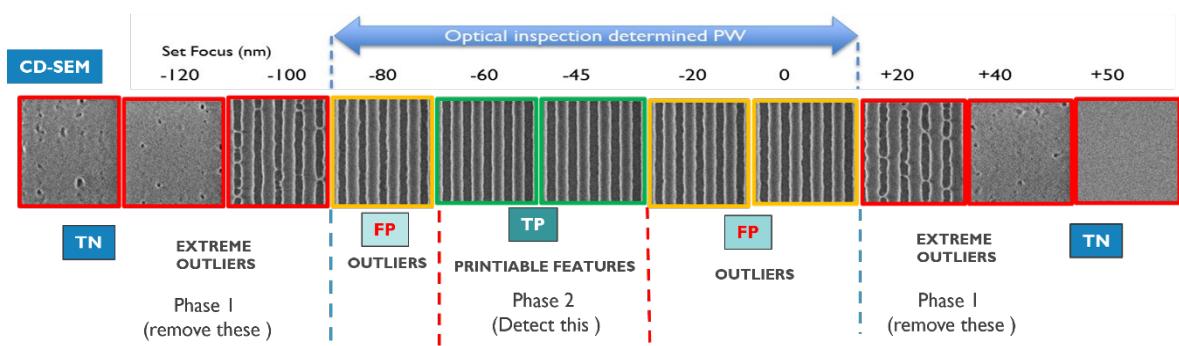
\*Part of the data reported in this chapter is reprinted with permission from Bappaditya Dey, Dorin Cerbu, Kasem Khalil, Sandip Halder, Philippe Leray, Sayantan Das, Yasser Sherazi, Magdy A. Bayoumi, Ryoung Han Kim, "Unsupervised machine learning based CD-SEM image segregator for OPC and process window estimation," Proc. SPIE 11328, Design-Process-Technology Co-optimization for Manufacturability XIV, 113281G (23 March 2020); <https://doi.org/10.1117/12.2552055>.

## **2.1. Introduction**

A semiconductor device such as an integrated chip is produced by a sequence of process steps. Many of them require a lithographic mask through which 1 or 2D pattern is printed on a photoresist layer, followed by one or more etch processes for transferring the pattern to the underlying layers. The design of such lithographic masks nowadays involves the prediction of the printed pattern considering optical, resist and etch phenomena, which occur at the scale of the mask below. These effects lead to unavoidable deviations on the printed pattern with respect to the design intent. Dedicated EDA tools, as an example OPC (Optical Proximity Correction) software are used for determining and making corrections to the intended initial mask design to provide best possible approximation of the design intent on the printed wafer. OPC optimization is based on the fragmentation of the initial design edges to compensate for the given phenomena and relies on an accurate modelling to predict simulated contours of the printed features. The window defines the printability performance limits in terms of the focus and exposure dose settings of a lithographic tool, within which limits a reliable print of the pattern is obtainable. Some patterns can have bigger printable windows than other patterns, due to various reasons such as design geometry, accuracy of the OPC modelling etc. The simulated windows are subsequently verified experimentally by manufacturing a test mask and using it to print various pattern features on a plurality of die areas of a photosensitive resist layer. Each die area is being printed with varying values of focus and exposure conditions. The process windows are determined by measuring features/CD on the plurality of die areas. Differences between the experimental and simulated windows are evaluated and may necessitate further OPC-based optimization of the mask design.

Different measurement techniques are known for analyzing the printed dies. Optical inspection may be used but as the feature dimensions shrink to critical dimensions under

32 nm, these techniques are not sufficiently accurate because optical tools cannot resolve the structures. Therefore e-beam tools such as CD-SEM (Critical Dimension-Scanning Electron Microscope) are used to obtain detailed images and the printable window is derived from a manual inspection of these images. This is however a time-consuming effort, and it represents a bottleneck that slows down the mask development process. Within the printable window, a narrower process window may be determined, related for example to specific defects, i.e., defects which occur when the dose and focus are outside the limits of the process window. The determination of this narrow process window today takes place primarily through the manual comparison of CD-SEM images and is therefore equally a labour intensive and time-consuming effort. The printable window and the defect-related process window are examples of process limits needed for evaluating an OPC based mask design and/or for correctly printing a given set of features on a semiconductor wafer. Process limits are determined not only in terms of dose and focus applied in lithography processes, but also as a function of various other process parameters, such as etch parameters or parameters related to deposition or planarization processes. Today, the determination of these process limit is based only on CD measurement and not completely on pattern fidelity.



**Figure 2.1.** Optical inspection determined Process Window. Reprinted with permission from [33].

Our proposed approach provides an alternative automated method for determining process limits that does not suffer from the above-described problems. We have divided the entire problem statement into 2 phases. In the first phase, our aim is to remove the extreme outliers (TNs, True Negatives) or non-printable features as depicted in Fig. 2.1 for pre-processing OPC data (data clean up). In the second phase, our aim is to exploit the same approach to automatically define defect process windows by providing a significant decision boundary between TPs (True Positives) and most importantly FPs (False Positives), where TPs are the printable features and TNs are non-printable features or erroneous features.

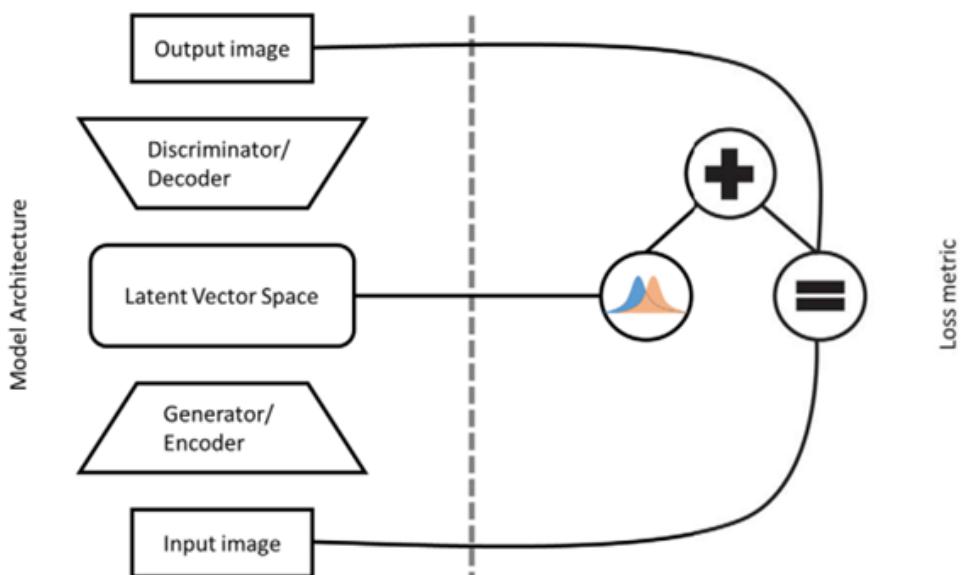
We have proposed a new method of analyzing the process window with an unsupervised machine learning approach. Our proposed algorithm will extract the underlying key features and encode these to latent feature vectors or latent vector space instead of the conventional CD, given a dataset of thousands of CD-SEM images, and then rank the images based on a similarity index and then to automatically determine the process window. The novelty of this idea lies in the fact that (1) with a defect inspection tool this task seems tedious and time consuming and often require human intervention to analyze a large number of features, (2) a CD-SEM based process window analysis might always not match with a defect inspection-based process window. Our generalized variational auto-encoder based approach does this automatically. Also, we have analyzed and validated our result against conventional tool approach, which is only CD-based. For 2D patterns this method becomes even more powerful.

## 2.2. Related Work

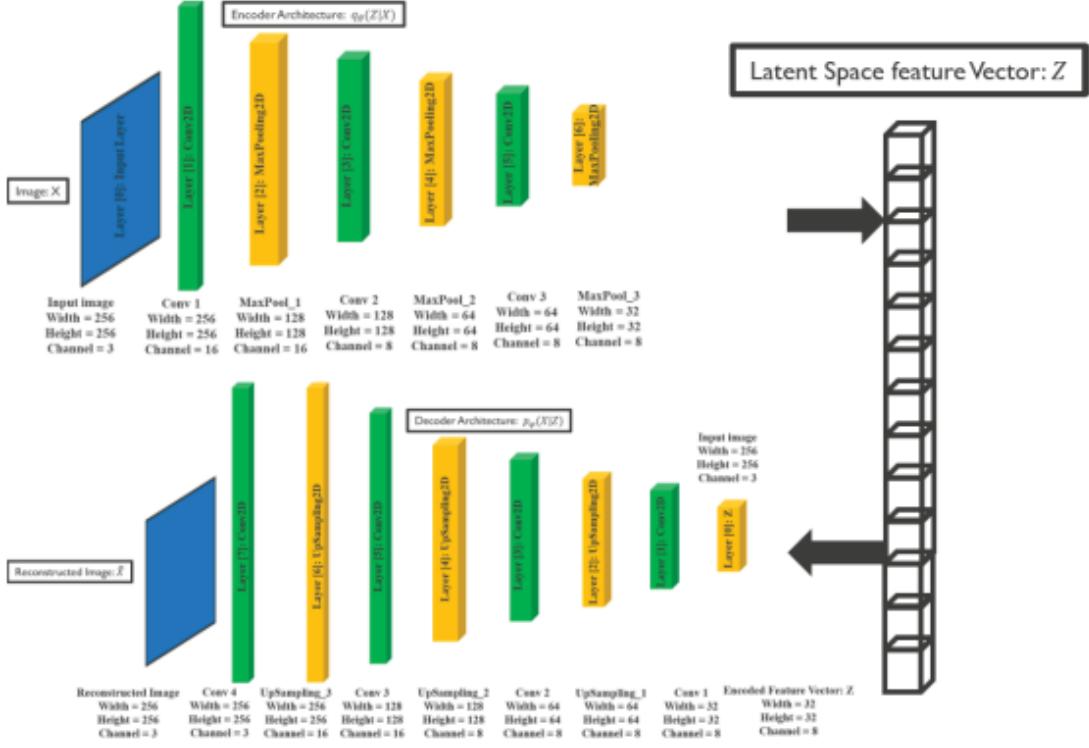
We briefly discussed existing research approaches and methodologies in the context of defect process window monitoring and the impact of EUV stochastics on the overall process window. Halder et. al. had carefully investigated the scope of performing future analysis with utilization of existing enormous review-SEM images for quantification with

CCA and K-means algorithms [13, 14]. In this research work, the authors have compared the full map of SEM overlay with a full map of optical overlay for high order correction in the context that CD-SEM could estimate asymmetry property of target from an image of pattern edge. They also evaluated the difference metrics between the via-pattern and relaxed pitch gratings [15]. Bisschop et. al. in his research work [16, 17] had presented a quantification-metric for stochastic failure and investigated experimental parameters dependency as well as presented a reasonable analysis for prediction of stochastic CD-variability in the context of OPC-verification. Halder et. al. had proposed an innovative methodology to improve accuracy for evaluating process window on wafer. They have adopted “sectorization” technique to differentiate different AOI for loss-less analysis in the context of CD-SEM images [18]. In this paper [19], the authors have discussed the context of higher defect detection sensitivity to identify precise process window and capabilities for automatic sub-categorization of defects. This research paper [20] demonstrated techniques to improve OPC metrology flow to optimize statistical process variation present in modern fabrication process flow.

### 2.3. Proposed Method



**Figure 2.2.** Generative Var-AE Model architecture. Reprinted with permission from [33].



**Figure 2.3.** The Encoder model compresses image data into a latent vector space ( $Z$ ). The Decoder model reconstructs the image ( $\hat{X}$ ) given the learned hidden representations. Reprinted with permission from [33].

Generative adversarial networks (GAN) are unsupervised or semi-supervised deep neural net architectures comprised of two architectures as shown in Fig. 2.2, competing one against the other [12], [21-24]. A Variational-Autoencoder [12], [25, 26] in generative-adversarial network architecture perspective, consists of 2 different sub-architectures as (1) a generator/encoder and (2) a discriminator/decoder and a trade-off metric between these two as a loss-function. The encoder model is termed as a “bottleneck” as it must learn an efficient and significant compression of the data into its corresponding lower dimensional stochastic space. The VAE-autoencoder architecture as shown in Fig. 2.2 can be summarized as follows:

1. An input image is passed through an encoder network.

2. The encoder outputs parameters of a distribution  $q_\theta(Z|X)$ , a Gaussian probability distribution.
3. A latent vector  $Z$  is sampled from  $q_\theta(Z|X)$ . If the encoder learned to do its job well, most chances are  $Z$  will contain the information describing  $X$ .
4. The decoder decodes  $Z$  into an reconstructed image  $\hat{X}$ . On the R.H.S we have the loss:
  - a. Reconstruction error: the output should be similar to the input. Difference between  $q_\theta(Z|X)$  and  $p_\phi(X|Z)$ .
  - b.  $q_\theta(Z|X)$  should be similar to the prior  $p(Z)$  (multivariate standard Gaussian).
5. Loss function term with Kuulback-Leibler divergence as:

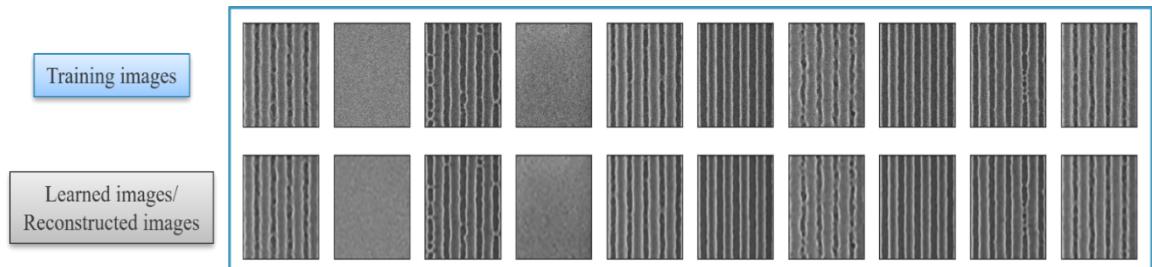
$$L(\theta, \varphi) = -\text{Eq}_\theta(Z|X)[\log p_\varphi(X|Z)] + KL(q_\theta(Z|X) \parallel p(Z))$$

We train the variational autoencoder using gradient descent to optimize the loss w.r.t parameters  $\theta$  and  $\varphi$ .

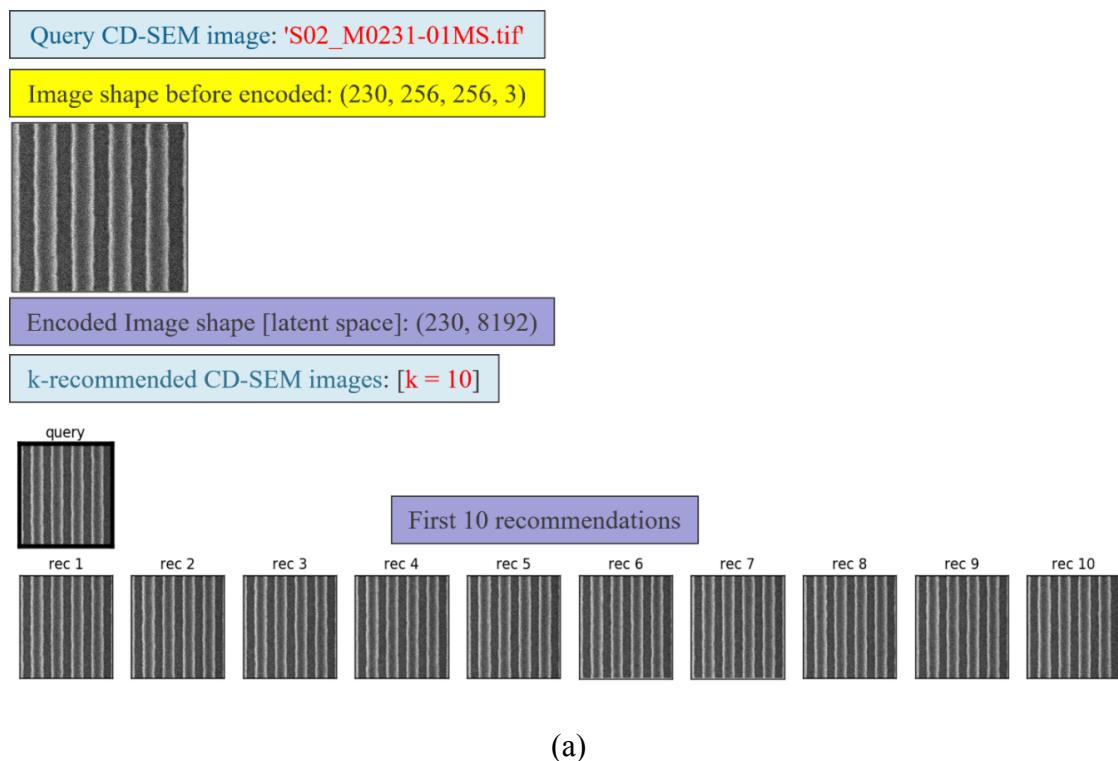
In order to generate new images, we can directly sample a latent vector from the prior distribution and decode it into an image. Our VAR-autoencoder architecture has an Encoder/Generator model with 3 pairs of convolutional-maxpool layers. The Decoder model has 4 convolutional layers and 3 up-sampling layers. We have used a convolutional kernel and a pool kernel. The encoder encoded each input image of size (*Width* = *Height* = 256, *Channel* = 3) into a latent space vector of size (1, 8192). The decoder took the learned

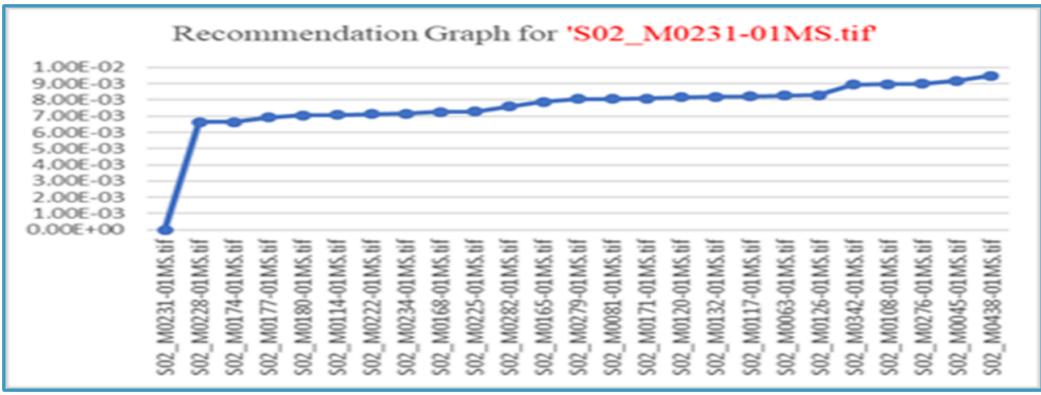
internal representations as latent space vector of that image using the learned weights and reconstructed the image back as shown in Fig. 2.3. We have used “ReLU” as activation function and kept the padding as “same”. We have a total of 4,963 trainable parameters.

## 2.4. Implementation and Experimental Results



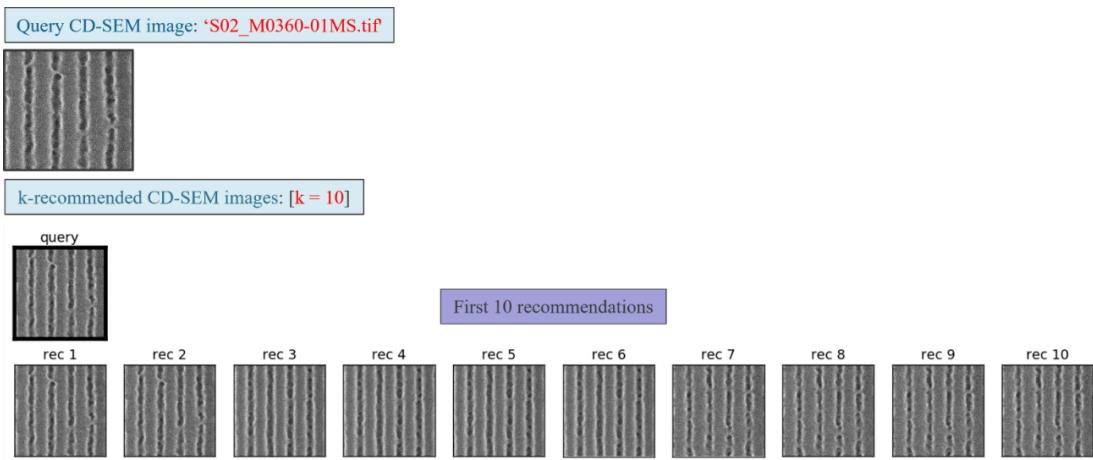
**Figure 2.4.** (a) Training images, (b) Learned/Reconstructed images by the model.  
Reprinted with permission from [33].



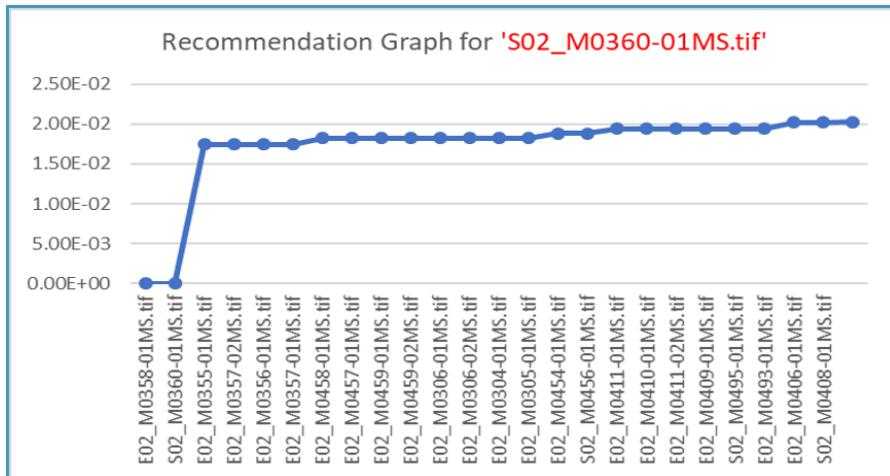


(b)

**Figure 2.5.** (a) Classification and Recommendation for CD (line-space) image pattern at dose: 42.0, focus: -0.09. The first 10 best recommended similar printable CD-SEM patterns (images) has been shown out of 230 diverse datasets of CD-SEM images. (b) Recommendation graph-plot for “query-pattern” index S02-M0231-01MS.tif. Reprinted with permission from [33].



(a)



(b)

**Figure 2.6.** (a) Classification and Recommendation for CD (line-space) image pattern at dose: 42.0, focus: -0.09. The first 10 best recommended similar non-printable CD-SEM patterns(images) has been shown out of 230 diverse datasets of CD-SEM images. (b) Recommendation graph-plot for “query-pattern” index S02-M0360-01MS.tif. Reprinted with permission from [33].

Our proposed approach is based on 2 major steps. In the 1st step, we have trained a prototype of Conventional Variational-AE [25, 26] with a set of 230 FEM wafer images for training and testing purposes. The dataset was divided into a training set, a validation set and a test set with a percentage ratio of [60: 20: 20]. The training set consisted of 187 CD-SEM images and the test set consisted of 43 CD-SEM images for different dose and focus values. We took the validation set by random shuffling the CD-images both from test and train set for each epoch. We also implemented data-augmentation techniques (rotation at 90 degrees, rotation at finer angels etc.) to balance/increase the diversity of training dataset image patterns. The model was trained with Intel Core *i5* – 8350U CPU 1.70GHz node. The VAR-AE model was trained using Keras library [27] and the Tensorflow library [28] backend in the python programming environment. The Anaconda version was 4.6.8. We reiterated the training process with variable batch-size ranging from 8 to 256 [29]. All CD-SEM images were normalized to have values in the range [0, 1]. We choose Adam optimization technique as stochastic gradient decent algorithm and binary-cross-entropy as loss-metric [30, 31]. Once training step has been completed with minimum reconstruction loss, we saved the model and learned weights for later application. We have extracted the saved learned encoder part of the model to extract/generate the representative features or embedded characteristics of the CD-SEM images (latent vector space encoding) during inference as shown in Fig 2.4.

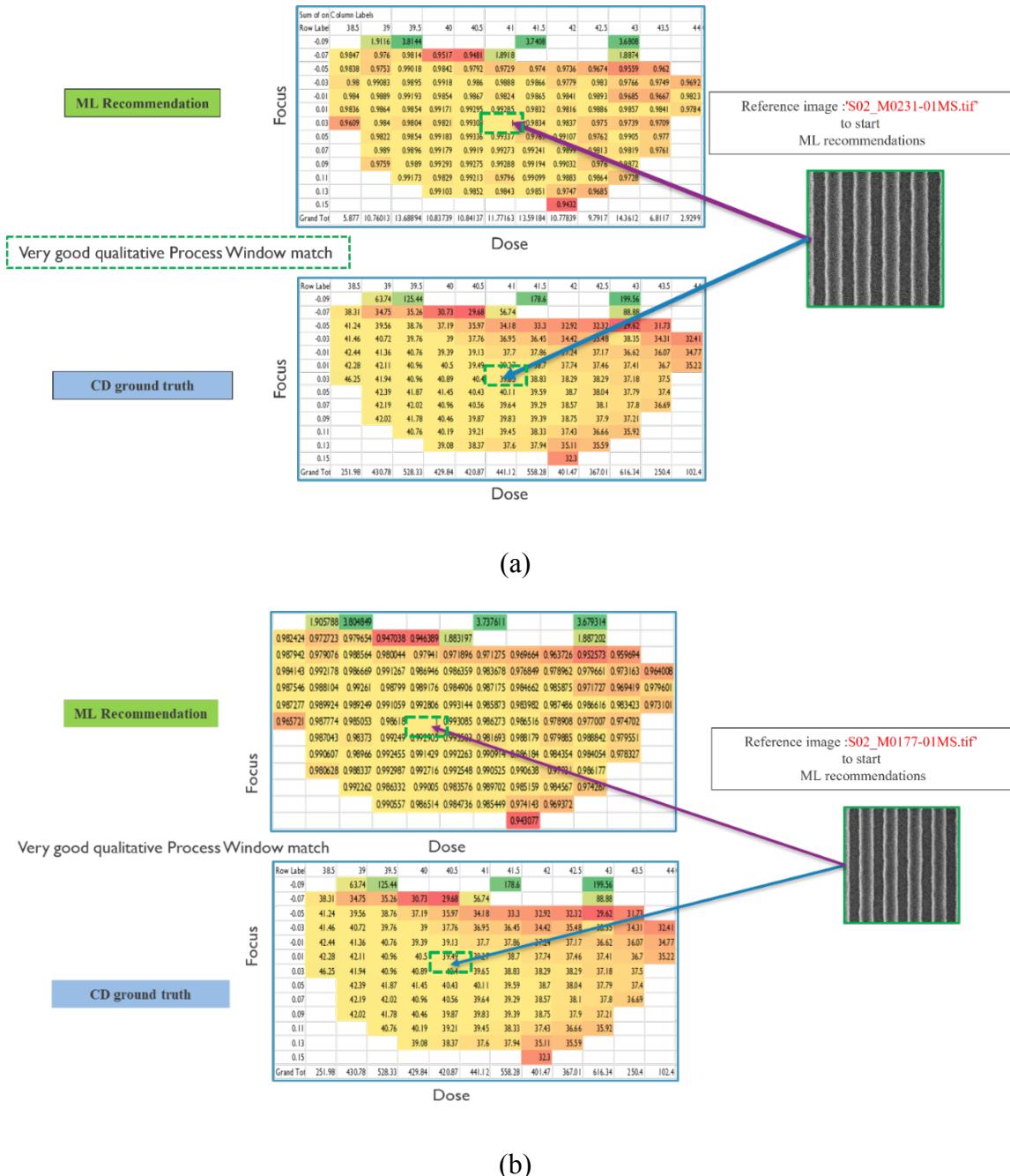
**Table 2.1.** Normalized distance metric for 5 different distance function for randomly selected 10 CD-SEM images. Reprinted with permission from [33].

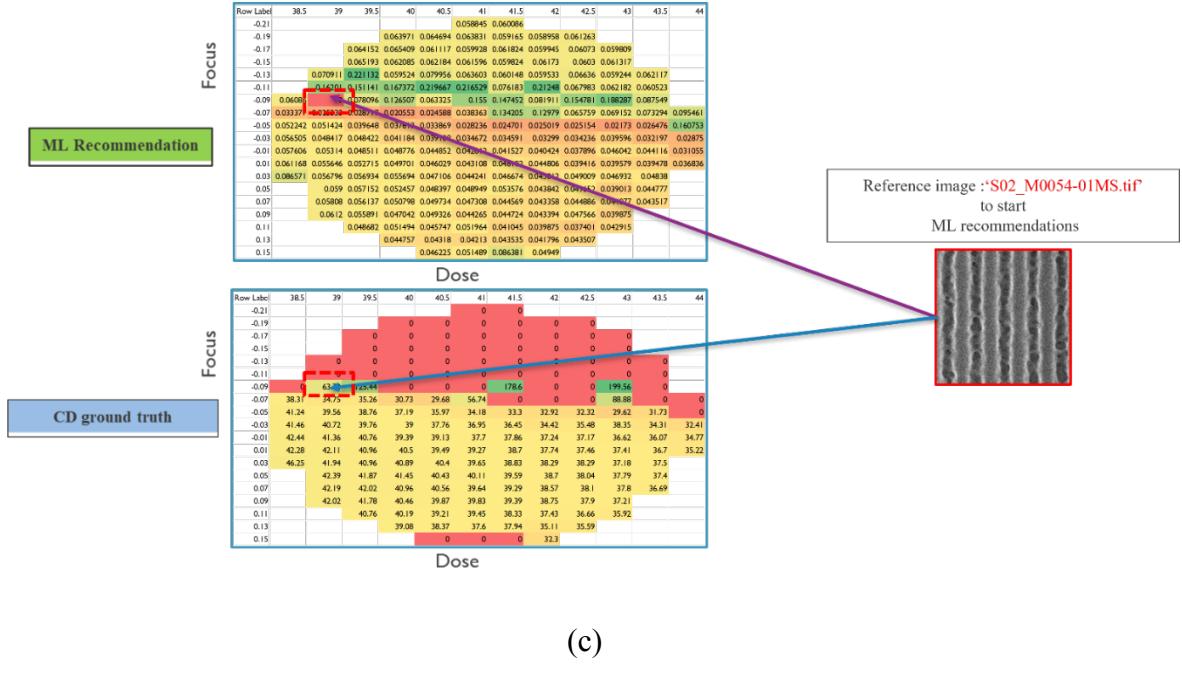
Image_name	Euclid_dist_norm	Manhattan_dist_norm	Jaccard_dist_norm	Cos_sim_dist_norm	Minkow_dist_norm
E02_M0022-01MS.tif	1.014483	1.074927	1.000136	1.008405	0.980607
E02_M0052-01MS.tif	1.046246	1.060798	1.000136	1.00414	1.01445
E02_M0055-01MS.tif	1.047701	1.086538	1.000136	1.013524	1.004681
E02_M0056-01MS.tif	1.047701	1.086538	1.000136	1.013524	1.004681
E02_M0057-01MS.tif	1.047701	1.086538	1.000136	1.013524	1.004681
E02_M0057-02MS.tif	1.047701	1.086538	1.000136	1.013524	1.004681
E02_M0058-01MS.tif	1.008539	1.053082	1.000136	1.013524	0.986324
E02_M0091-01MS.tif	1.036588	1.045348	1.000136	1.005846	1.012918
E02_M0092-01MS.tif	1.036588	1.045348	1.000136	1.005846	1.012918

**Table 2.2.** Normalized Ground-Truth Mean/CD obtained from conventional techniques. Reprinted with permission from [33].

Dose	Focus	Mean_CD_norm	Delta_CD
39	-0.09	0.830597	-0.20276
39.5	-0.09	0.817305	-0.21605
39.5	-0.09	0.817305	-0.21605
39.5	-0.09	0.817305	-0.21605
39.5	-0.09	0.817305	-0.21605
41	-0.07	0.73938	-0.29398
41.5	-0.09	1.16367	0.13031
41.5	-0.09	1.16367	0.13031
41.5	-0.09	1.16367	0.13031

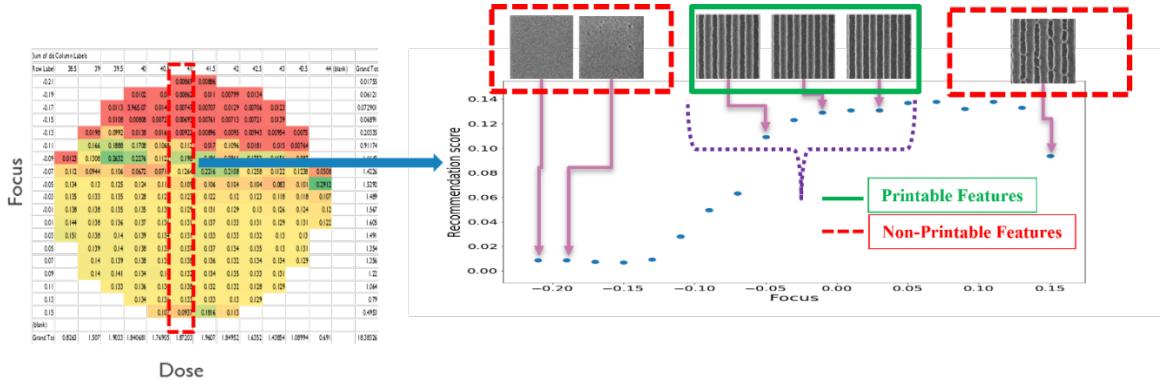
In the second step, we have performed similar pattern retrieval or similar CD-SEM image retrieval on given test database of images by implementing well-known KNN algorithm [32] on the above extracted encoded feature space. We have experimented with 5 popular different distance metrics as Cosine-Similarity, Minkowski Distance, Manhattan Distance, Euclidean distance and Jaccard Distance and found the context of choosing the Cosine-Similarity as distance metric for KNN learning fits the best as shown in Table. 2.1. We believe the training has been performed with limited resources and with very limited scope, so further careful analysis, and improvement of hyperparameters (both the model architecture and area-of-subject-research) may significantly improve the scope of training process. We have termed any test-case “Unlabelled” CD-SEM image patterns as a “Query image” or a “Query-pattern” against rest of unseen diverse dataset of other CD-SEM images/patterns and our proposed approach has ranked them based on most “closer” or “similar” learned encoded-feature metric in an unsupervised way of training. In other words, our proposed approach allows us to extract the process limits directly from the score/rank values rather than from a manual comparison of the images. Fig. 2.5 (a) and Fig. 2.6 (a) have shown 2 different sets of classification - [1: ( $N - 1$ )] recommendation pair for respective printable “query” CD-SEM image pattern (indexed here as S02\_M0231-01MS.tif) and non-printable “query” CD-SEM image pattern (indexed here as S02\_M0360-01MS.tif). The respective recommendation variations have been also depicted in Fig. 2.5 (b) and Fig. 2.6 (b). Fig. 2.7 [(a)-(c)] shows the wafer-map representation for 3



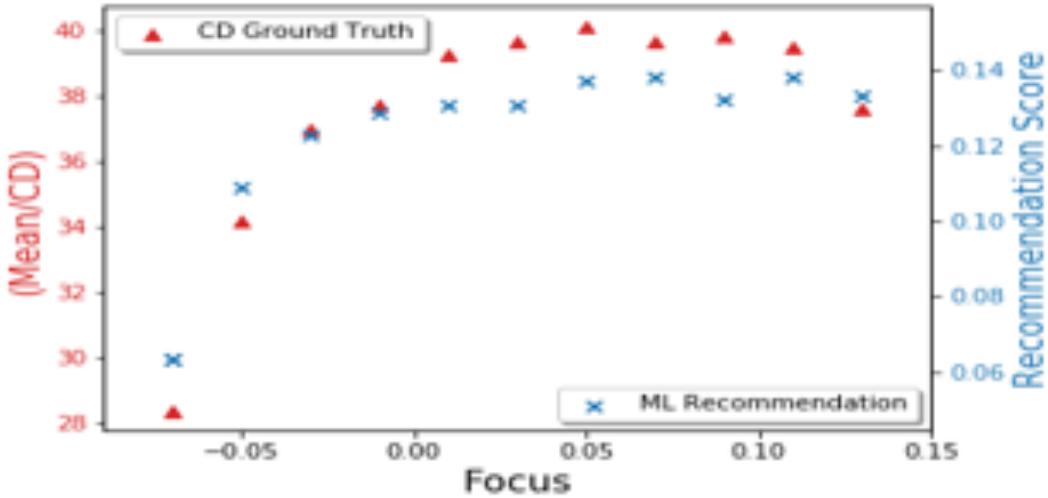


**Figure 2.7.** Wafer-map representation for Conventional technique metric vs. proposed ML recommendation for (a) a very good qualitative Process-Window (best printable feature CD-SEM image). (b) for a near good qualitative Process-Window (a random near best printable feature CD-SEM image). (c) for a worst qualitative Process-Window (random non-printable feature CD-SEM image). Reprinted with permission from [33].

different case-scenario, where we have extracted the process-limits directly from the obtained score-values/ranking metric and plotted against conventional CD-ground truth (obtained from conventional tool approach). Obtaining a score value representative of a given image characteristics is enabled by a targeted choice of one or more of the query images and object images, and/or by determining the score as an aggregate score value, calculated from the scores obtained with respect to a set of different “query” images. We have shown the ground-truth Mean/CD values obtained from the conventional tool approach for random 10 CD-SEM images in Table. 2.2. The aggregate score may be the maximum or the average of the scores of



**Figure 2.8.** Qualitative Process-Window analysis for a reference Dose-Column: 41.0 with varying focus. Reprinted with permission from [33].



**Figure 2.9:** Plot for Normalized Mean/CD-ground truth value against proposed ML-Recommendation score for the selected Dose column: 41.0 mJ/cm<sup>2</sup> against varying focus. Reprinted with permission from [33].

the object images, obtained with respect to a set of “query” images. From Fig. 2.8 and Fig. 2.9, supports our claim that our proposed unsupervised machine learning approach can classify and segregate printable CD-SEM images/patterns from the non-printable CD-SEM images/patterns and extract the probable process limits directly from the score values, rather than from a manual comparison of the images. Fig. 2.9 also depicts that the scores furthermore closely follow the variations of the CD beyond sharp rise. This result shows that it is possible to determine a CD-related process-window (i.e., a narrower window

within the limits of the printable window) based on the score values alone, provided a preliminary calibration is performed.

## 2.5. Summary

The proposed method provides a way of automatically detecting the probable printable window. We have merged concepts of unsupervised learning and Deep-Generative-Network to first learn the characteristic features for a given dataset of numerous different CD-SEM images and then ranked them based on a numeric score-metric, derived from comparison of the images in the latent feature space ( $1:N - 1$ ). When the query image is chosen judiciously, the score of the object images exhibits a monotonic relation to CD of the images. The method becomes more relevant for complex  $2D$  shapes/patterns where only CD-measurement is not enough. Hence, our proposed deep unsupervised learning approach offers an effective method for deriving the printable window from the score values, without necessitating the actual CD-measurements, by detecting the presence of a step in the relation between the score and the focus and/or dose. Our future goal is to improve the model accuracy and performance by fine-tuning different hyper-parameters to clearly distinguish small variations by sensing context-pixel differences only and to look for stochastic defects during review of defect locations.

## **Chapter 3: SEM Image Denoising With Unsupervised Machine Learning for Better Defect Inspection and Metrology\***

CD-SEM images inherently contain a significant level of noise. This is because a limited number of frames are used for averaging, which is critical to ensure throughput and minimize resist shrinkage. This noise level of SEM images may lead to false defect detections and erroneous metrology. Therefore, reducing noise in SEM images is of utmost importance. Both conventional noise filtering techniques and recent most discriminative deep learning based denoising algorithms are restricted with certain limitations. The first enables the risk of loss of information content and the later mostly requires clean ground-truth or synthetic images to train with. In this chapter, we have proposed a U-Net architecture based unsupervised machine learning approach for denoising CD-SEM images without the requirement of any such ground-truth or synthetic images in true sense. Also, we have analysed and validated our result using **MetroLER, v2.2.5.0.** library. We have compared the power spectral density (PSD) of both the original noisy and denoised images. The high frequency component related to noise is clearly affected, as expected, while the low frequency component, related to the actual morphology of the feature, is unaltered. This indicate that the information content of the denoised images was not degraded by the proposed denoising approach in comparison to other existing approaches.

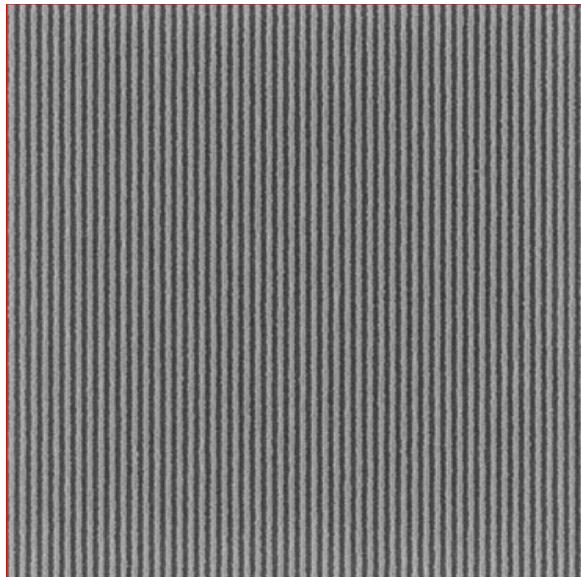
### **3.1. Introduction**

Scanning Electron Microscope (SEM) are used widely in the semiconductor industry for metrology and inspection. Among the various types of SEMs, CD-SEMs are of profound importance mainly because they measure the CD (critical dimension) of the circuit patterns based on which the entire litho process is targeted. Review-SEMs and E-beam inspection

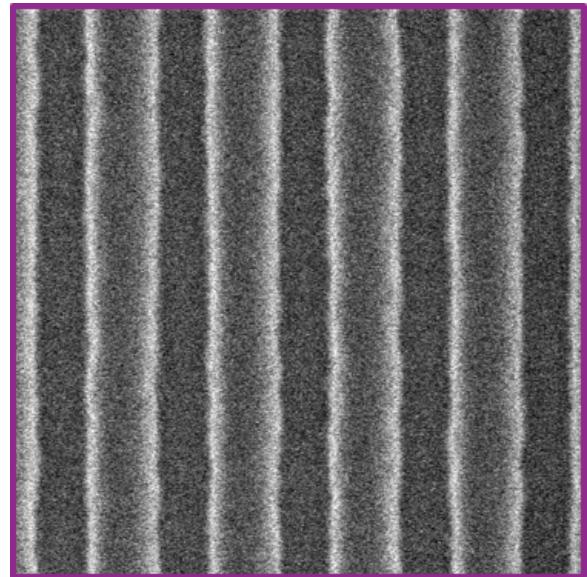
---

\*Part of the data reported in this chapter is reprinted with permission from Bappaditya Dey, Sandip Halder, Kasem Khalil, Gian Lorusso, Joren Severi, Philippe Leray, Magdy A. Bayoumi, "SEM image denoising with unsupervised machine learning for better defect inspection and metrology," Proc. SPIE 11611, Metrology, Inspection, and Process Control for Semiconductor Manufacturing XXXV, 1161115 (22 February 2021); <https://doi.org/10.1117/12.2584803>.

tools are gaining importance as we shrink to N3 nodes and below because of their high resolution. However, as circuit patterns become smaller (pitches less than 32 nm) the extraction of repeatable and accurate defect locations along with CD metrology becomes significantly complicated especially post ADI (After Develop inspection). This is simply because at these pitches the number of pixels available to detect a defect or do metrology is approaching single digit numbers. Hence, reducing noise in SEM images is of utmost importance.



(a)



(b)

**Figure 3.1.** Noisy CD-SEM images (a) Pitch 32 and (b) Pitch 90. Reprinted with permission from [56].

The case of ADI SEM imaging is one of the most challenging tasks. Images are usually noisy when less number of frames are used. This often leads to false defect detections and erroneous metrology. Fig. 3.1 shows example of noisy SEM images for two different pitches. Detection of minute bridges and breaks and resist footing from these top-down images is becoming more and more important. While there are various noise-removal techniques available today not all of them survive the requirements of the semiconductor industry at these advanced nodes. Also, the quantification of noise for SEM images does not fit truly Gaussian or Poisson distributions [34], as noise bias varies with metrology settings and sample properties. Recent discriminative deep learning-based algorithms have outperformed the conventional noise filtering methods. However, these methods need to be trained with clean ground-truth images which in most scenarios do not exist. As an alternative, some methods have used synthetic noiseless images generated from software-tools for Supervised methods or have degraded the clean target with approximated noise level for Semi-Supervised methods. Often these methods require additional conditional files to generate the corresponding synthetic images, and this may lead to an additive artefacts. Machine learning is significant in this issue. Currently, machine learning is significant for learning patterns for classification. Machine learning methods have been used in several domains such as speech recognition [35], hardware fault prediction [36], natural language processing [37], embryonic hardware system [38], image recognition [39]. Machine learning is divided into supervised and unsupervised learning. In supervised learning, variables of input and output are given. The training is done using labelled data such as neural network [40-42], support vector machine [43], recurrent neural network [44], linear and logistics regression, etc. In unsupervised machine learning, variables of input only are given. The learning is done without labelled data such as K-means [45], Cluster

algorithms [46], etc. We have applied a U-Net architecture based unsupervised machine learning approach for denoising SEM images without the requirement of any such ground-truth or synthetic images. This chapter addresses the following problems: (1) Denoising SEM images without clean target in true sense. The proposed method can be trained using only a single or few noisy SEM images. (2) Robustly detect the feature-edges from denoised images using unsupervised method for better metrology and defectivity inspection. The approach is tested for both simulated and real SEM images. Also, we have analysed and validated our result against conventional approach. Our proposed unsupervised approach demonstrates its effectiveness both quantitatively and qualitatively.

### 3.2. Related Work

We briefly discussed some existing recent research approaches and methodologies in the context of SEM image denoising/quality enhancement techniques. Liangjiang Yu et al. [47] has proposed an unsupervised machine learning based SEM image quality enhancement framework as **uMLIQE**. This framework requires only noisy images. The authors captured noisy data pairs at random die locations without any frame averaging. The framework reportedly converges to a statistical mean of the desired target. The main drawbacks/limitations of this proposed method as reported are induced artifacts and contrast loss. In this paper, the authors have presented a deep learning-based method to realize accurate and stable CD measurement from SEM images with low SNR [48]. They constructed the dataset used in deep learning training using a CD-SEM image model designed from mathematical and statistical line edge roughness (LER). This approach also has limitations due to induced artifacts. D. Cerbu et al. [49] has also presented similar work on SEM image denoising for a better metrology and defect inspection. The authors have trained a generative network with noisy images. They have used noiseless synthetic images and then added Gaussian white uncorrelated noise to it. For the discriminator network, they

have used ideal/noiseless images. After training is completed, the generative network will act as a denoising filter. This is a nice approach to tackle noise in CD-SEM images but again limited with distortion/artifacts. N. Chaudhary et al. [50] proposed a deep supervised learning framework for the estimation of line edge roughness (LER) and line-width roughness (LWR) in low-dose scanning electron microscope (SEM) images. They have simulated a dataset of 100,800 SEM images constructed with the help of Thorsos method and the ARTIMAGEN library. Bappaditya Dey et al. [33] proposed an unsupervised deep learning-based approach to automatically detect the probable printable window. The proposed model enables to learn the characteristic features for a given dataset of numerous different CD-SEM images and then ranked them based on a numeric score-metric, derived from the comparison images in a N-dimensional latent feature space.

### 3.3. Proposed Method

The proposed denoising approach is based on Noise2Void training scheme introduced by Krull et al. [51]. A noisy image  $x$  can be represented as:

$$x = s + n \quad \text{Eq. 3.1}$$

Where,  $s$  is the signal part and  $n$  is the stochastic noise. Any image denoising mechanism aims to separate the signal degrading noise from the noisy signal to generate clean signal as  $[s = x - n]$  based on two assumptions: (a) pixel values in  $s$  are not statistically independent and (b) pixel values in  $n$  are conditionally independent given corresponding signal values.

In the context of noisy SEM images, we can summarize the above assumptions in mathematical form as:

- A joint probability distribution function  $p(s, n)$  to generate an image  $x$  as:

$$p(s, n) = p(s)p(n|s) \quad \text{Eq. 3.2}$$

- Satisfying (a) as

$$p(s_i | s_j) \neq p(s_i) \quad \text{Eq. 3.3}$$

for 2 pixels  $i, j$  with a Receptive-Field (RF)

- Satisfying (b) as

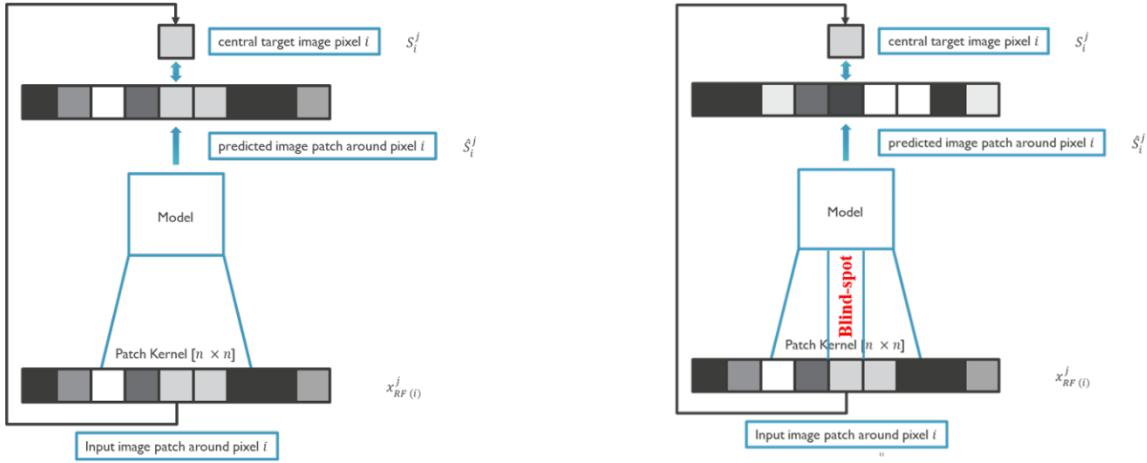
$$p(n|s) = \Pi_i p(n_i | s_i) \quad \text{Eq. 3.4}$$

$$E[x_i] = s_i \rightarrow gt[\text{noiseless image}] \text{ iff } E[n_i] = 0 \quad \text{Eq. 3.5}$$

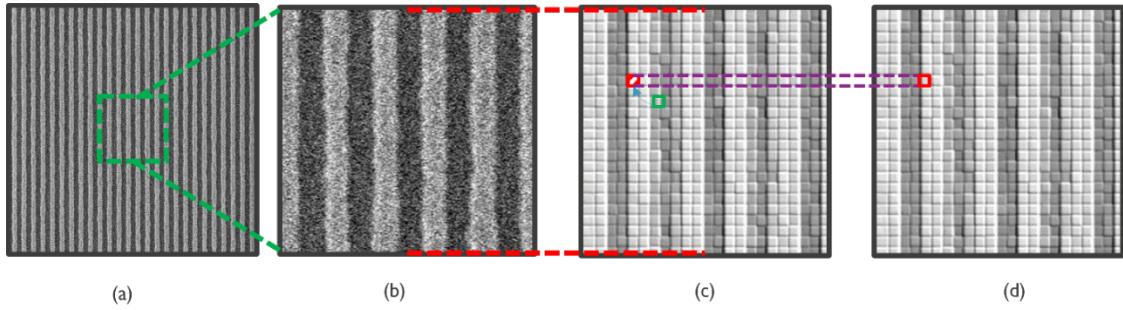
We have applied a U-Net architecture based unsupervised machine learning approach for denoising SEM images without the requirement of any such ground-truth or synthetic image. We have used a single noisy SEM image both as an input and target during training of our algorithm. The conventional and the proposed framework based on [51] have been shown in Fig. 3.2. The loss function can be represented as:

$$\arg_w \min \sum_j \sum_i J(g(\tilde{x}_{RF(i)}^j; w), x_i^j) \quad \text{Eq. 3.6}$$

where  $\tilde{x}_{RF(i)}^j$  is the receptive field of the model with blind-spot at the center as described in Fig. 3.3. Our goal is to minimize pixel-wise loss by fine tuning the parameters  $w$ . Due to the blind-spot mechanism the model is not able to learn the identity i.e., direct mapping from input patch to the target patch. Therefore, the model is able to estimate the signal part by analyzing the surroundings (statistical dependence) but carry no information about the noise part (statistically pixel-wise independent).



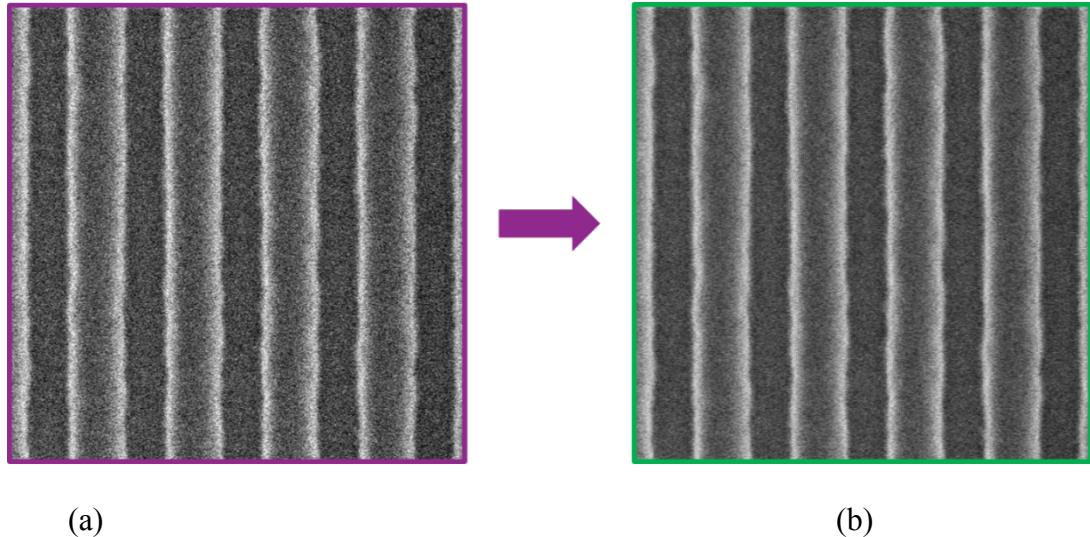
**Figure 3.2.** Conventional vs Proposed network. Reprinted with permission from [56].



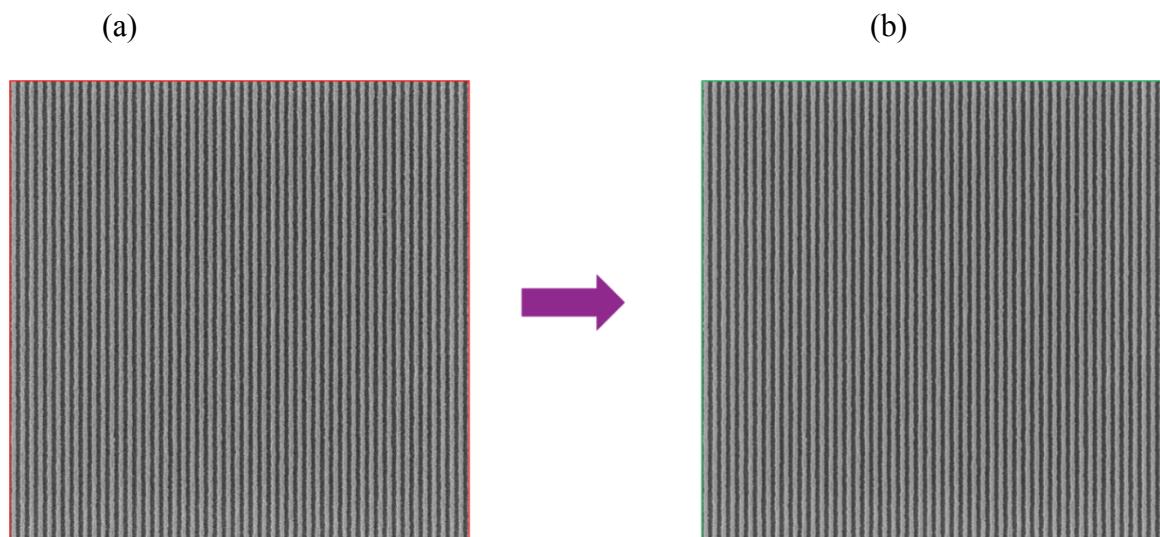
**Figure 3.3.** Blind-spot masking mechanism. (a) Original noisy SEM image. (b) image patch extracted from (a). (c) image is modified by replacing a target pixel (labelled red) with a randomly selected pixel value (labelled green) and thus creating a blind-spot. This modified image is used as input during model training. (d) target patch corresponding to (c) as original input patch without modification. Reprinted with permission from [56].

Our model has been trained on Lambda TensorBook with NVIDIA RTX 2080 MAX-Q GPU. We have randomly extracted patches from a single noisy SEM image of size  $64 \times 64$ . We have set number of training iterations in the range of 100 – 150 based on model convergence performance with number of trainings steps per epoch as 200 to 400. We have set the size of training batches as 128, depth of U-Net as 2, size of convolution kernels in first layer as 3, initial learning rate as 0.0001, number of feature channels in the first U-Net layer as 32. The patches/dataset was divided into a training set and a validation

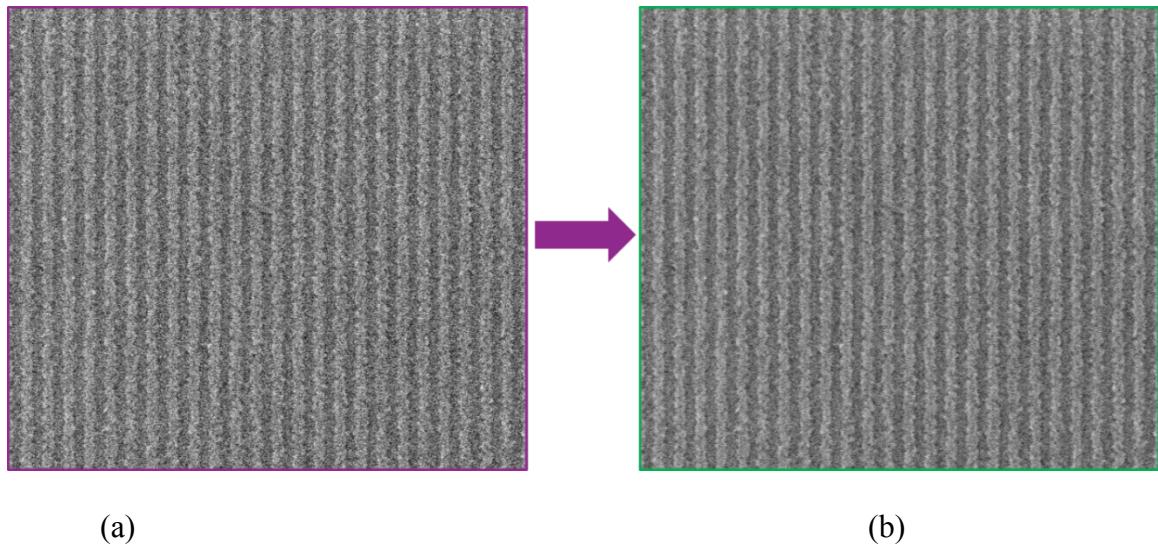
set with percentage ratio of [60: 40]. All the images are in TIFF format. The model was trained using Keras library [27] and the Tensorflow library [28] backend in the python programming environment. The Anaconda version was 4.6.8. Batch Normalization [29] is added to the U-Net architecture [52] before each activation function.



**Figure 3.4.** (a) Single noisy SEM image is given to the network model. (b) Denoised image generated by the network model without altering real signal. Reprinted with permission from [56].



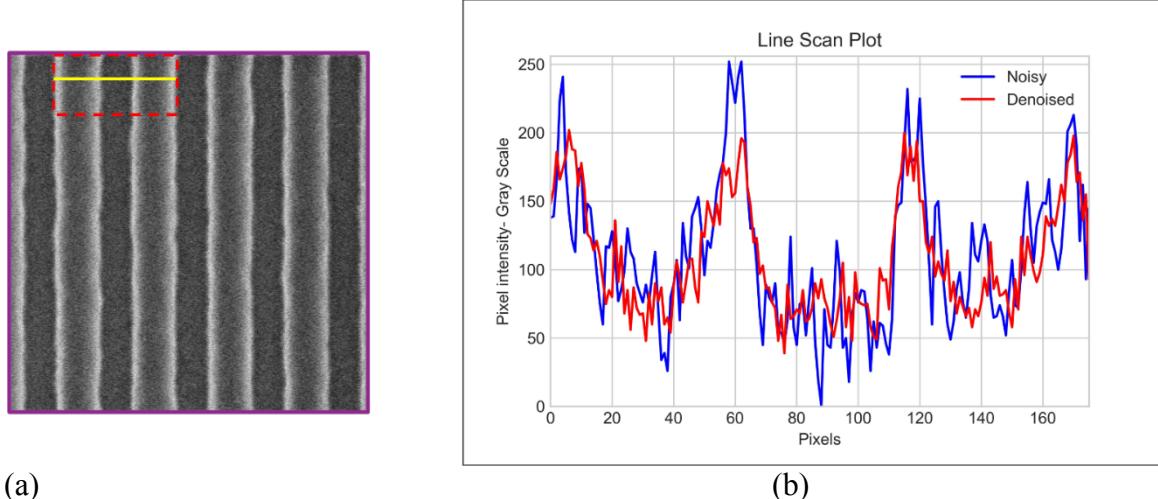
**Figure 3.5.** (a) Noisy SEM image P32 with Film thickness 30 nm (b) Denoised image. Reprinted with permission from [56].



(a)

(b)

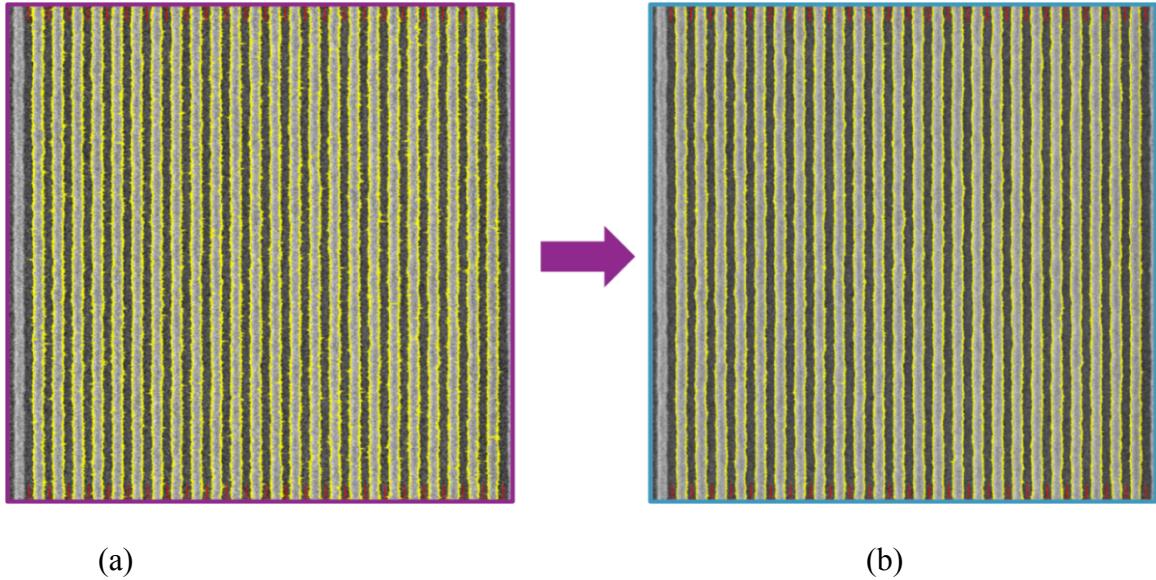
**Figure 3.6.** Cropped ROI for (a) Noisy SEM image P28 with Film thickness 8 nm [Thin Resist] (b) Denoised image. Reprinted with permission from [56].



(a)

(b)

**Figure 3.7.** (a) Original noisy SEM image ROI [P90]. (b) Line Scan Plot comparison for both noisy and denoised SEM images ROI. Reprinted with permission from [56].



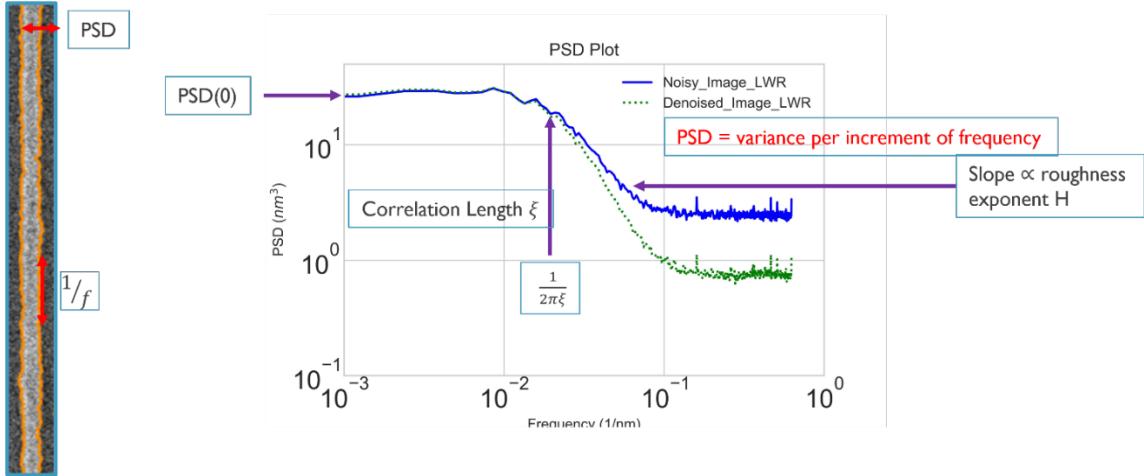
(a)

(b)

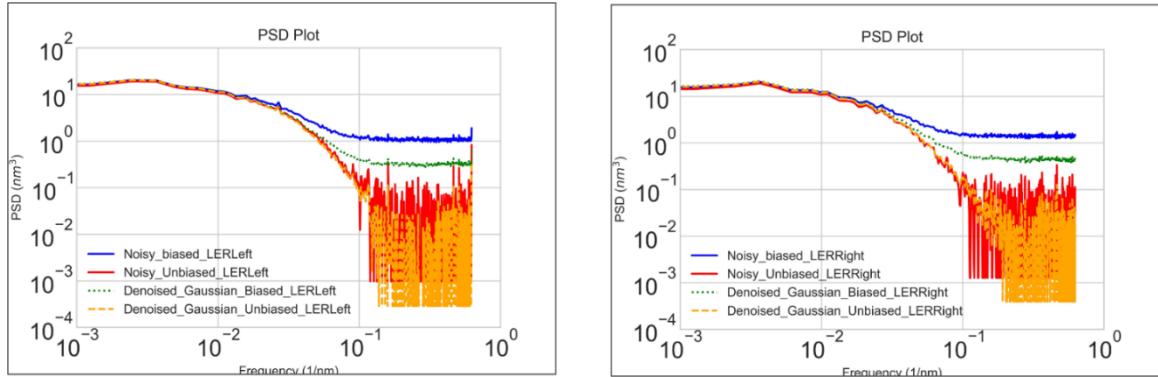
**Figure 3.8.** SEM image analysis with Fractilia MetroLER library (a) Noisy image (b) Denoised images. Reprinted with permission from [56].

### 3.4. Implementation and Experimental Results

Our proposed approach allows denoising of SEM images without requirement of any clean target (ideally noiseless) in true sense. In Fig. 3.4, Fig. 3.5, and Fig. 3.6, we have shown the results obtained from the proposed denoising approach for different pitches (as 90nm, 32nm, 28nm etc.) and with different film thicknesses (as 30 nm, 8 nm etc.). Fig. 3.7 shows the line-scan plots for both the noisy and denoised images. The two line-scans are overlapping one another, this indicates that the actual information content of the image was not degraded by the proposed denoising approach. Only noise part is removed as well as no contrast loss. We have analyzed each pair of original noisy image and its corresponding denoised image with Fractilia MetroLER v2.2.5.0. library [53]. Fig. 3.8 shows detected edges for denoised image are with less spikes or almost without spikes in comparison to the original one.



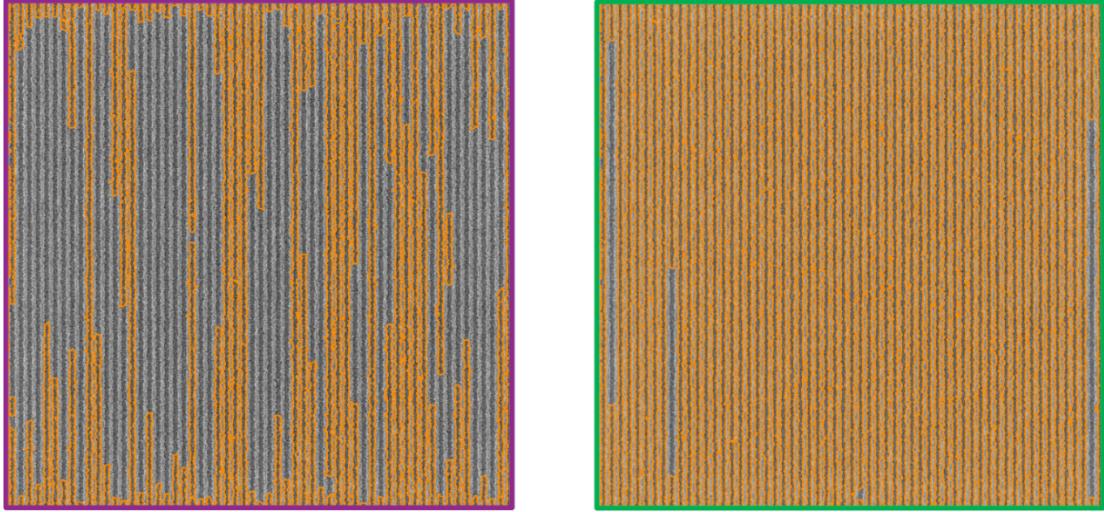
**Figure 3.9.** Power Spectral Density (PSD) plot for Line-Width-Roughness (LWR). Reprinted with permission from [56].



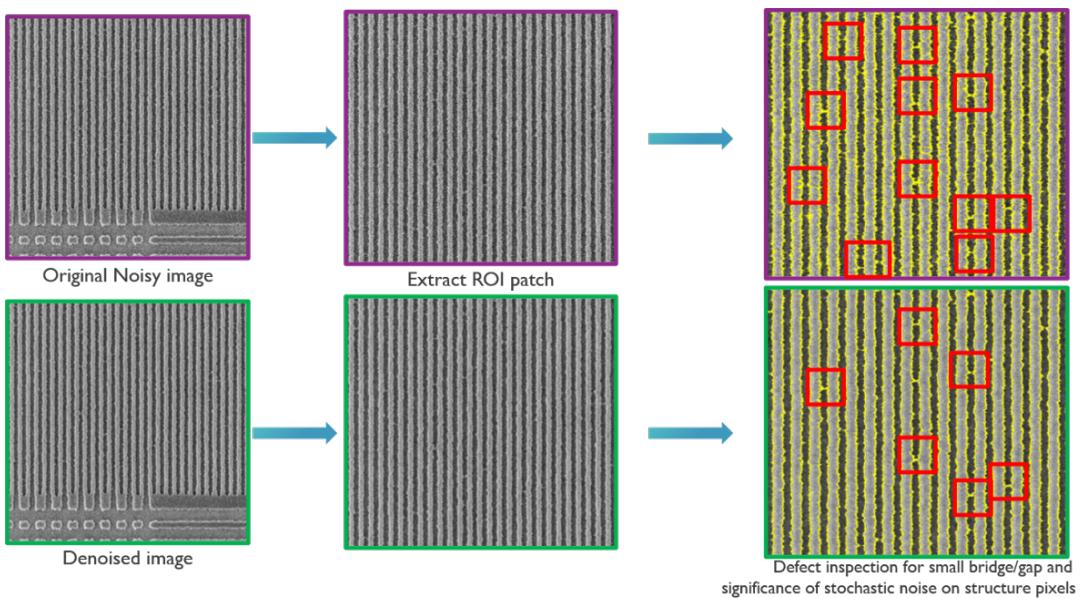
**Figure 3.10.** Power spectral density (PSD) plot for Line-Edge-Roughness (LER). Reprinted with permission from [56].

In Fig. 3.9, the power spectral density of a noisy and a denoised images are compared for Line-Width-Roughness (LWR). Power Spectral Density enables us to characterize the frequency behaviour of the roughness. The high frequency component related to noise is clearly affected, as expected, while the low frequency component, related to the actual morphology of the feature, is unaltered so that edge placement errors are not changing. This indicates that the information content of the images was not degraded by the proposed denoising approach. Fig. 3.10 depicts the Power Spectral Density (PSD) plot for Line-

Edge-Roughness (LER). For biased part, the noise level is lowered significantly whereas the unbiased part is unaffected post denoising.



**Figure 3.11.** Unsupervised GMM based CD/Contour extraction: (a) noisy vs (b) denoised for Thin resist low-ft images. Reprinted with permission from [56].



**Figure 3.12.** Defectivity analysis and accuracy metric comparison for Denoised image against Original noisy image. Reprinted with permission from [56].

We have applied the unsupervised GMM based segmentation method to extract the Critical Dimension (CD) or contour in noisy SEM images [54, 55]. Fig. 3.11 shows, for noisy images we are missing the granularity for major pixels and thus cd extraction is failing whereas denoised image makes better CD extraction possible with same parameter settings

for the algorithm in both scenarios. Therefore, this proposed deep learning based denoiser framework may be utilized as assist tool for defect inspection and better metrology. Our proposed approach also allows us to work with thin resists. Fig. 3.12 supports our claim that proposed denoiser as an assist tool for defect inspection. We have shown a comparative analysis for defect inspection for small bridge/gap and significance of stochastic noise on structure pixels for both noisy and denoised images. For noisy images, certainly more False Positive (FPs) defects are flagged due to the presence of stochastic noise on the body of SEM images. Therefore, leads to erroneous metrology which in turn leads to shape fidelity loss and finally yield loss. Whereas, for denoised images, we can correctly detect the defects (TPs) as stochastic noise has been removed or optimized.

### 3.5. Summary

The proposed method presented an unsupervised machine learning based SEM image denoising technique without the requirement of any ground-truth or synthetic images. The deep learning framework is based on a U-Net architecture. We have compared the power spectral density (PSD) of both a noisy and denoised image. The high frequency component related to noise is clearly affected, as expected, while the low frequency component, related to the actual morphology of the feature, is unaltered. This indicates that the information content of the denoised images was not degraded by our proposed denoising approach. We have also implemented unsupervised GMM based pattern classification and segregation technique to robustly detect feature edges from the denoised images. This approach helps to overcome the limitations of conventional edge-detection algorithms like Canny-edge detector in terms of parameter settings. This deep learning based denoiser framework may allow us to better identify defects as well as to work with thin resists. Our future goal is to improve the model accuracy and performance by using an arbitrary noise model obtained from a set of SEM images subject to uniform noise type.

## **Chapter 4: Deep Learning-Based Defect Classification and Detection in SEM Images \***

Defect inspection in semiconductor processes has become a challenging task due to continuous shrink of device patterns (pitches less than 32 nm) as we move from node to node. Current state-of-the-art defect detection tools (optical/e-beam) have certain limitations as these tools are driven by some rule-based techniques for defect classification and detection. These limitations often lead to misclassification of defects, which leads to increased engineering time to correctly classify different defect patterns. In this chapter, we propose a novel ensemble deep learning-based model to accurately classify, detect and localize different defect categories for aggressive pitches and thin resists (High NA applications). We train RetinaNet models using different ResNet, VGGNet architectures as backbone and present the comparison between the accuracies of these models and their performance analysis on SEM images with different types of defect patterns such as bridge, break and line collapses. Finally, we propose a preference-based ensemble strategy to combine the output predictions from different models in order to achieve better performance on classification and detection of defects. The proposed ensemble method demonstrates improvement of the average precision metric (mAP) of the most difficult defect classes. As CD-SEM images inherently contain a significant level of noise, detailed feature information is often shadowed by noise. For certain resist profiles, the challenge is also to differentiate between a microbridge, footing, break, and zones of probable breaks. Therefore, we have applied an unsupervised machine learning model to denoise the SEM images to

---

\*Part of the data reported in this chapter is reprinted with permission from Bappaditya Dey, Dipam Goswami, Sandip Halder, Kasem Khalil, Philippe Leray, Magdy A. Bayoumi, “Deep Learning-based Defect Classification & Detection in SEM Images.” [Online]. Available: <https://slideslive.com/38965533/deep-learningbased-defect-classification-detection-in-sem-images>

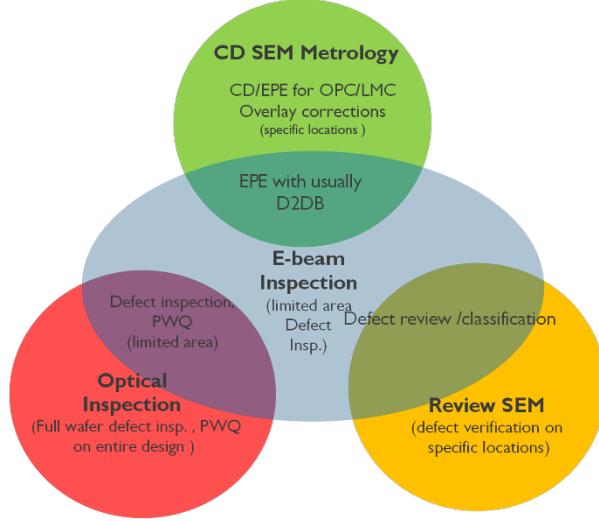
\*Part of the data reported in this chapter is reprinted with permission from E. Dehaerne, B. Dey, and S. Halder, “Ensemble deep learning-based defect classification and detection in sem images.” [Online]. Available: Ensemble Deep Learning-based Defect Classification and Detection in SEM Images ([learnopencv.com](http://learnopencv.com))

remove the False-Positive defects and optimize the effect of stochastic noise on structured pixels for better metrology and enhanced defect inspection. We repeated the defect inspection step with the same trained model and performed a comparative analysis for “robustness” and “accuracy” metric with conventional approach for both noisy/denoised image pair. Finally, we have demonstrated the application of Mask-RCNN (Regional Convolutional Neural Network), a deep-learning algorithm for computer vision and specifically object detection, to semiconductor defect inspection domain. We have extended our work for improved defect instance segmentation in SEM images with precise extent of defect as well as generating a mask for each defect category/instance. This also enables to extract and calibrate each segmented mask and quantify the pixels that make up each mask, which in turn enables us to count each categorical defect instances as well as to calculate the surface area in terms of pixels. Our proposed methods demonstrate its effectiveness both quantitatively and qualitatively.

#### **4.1. Introduction**

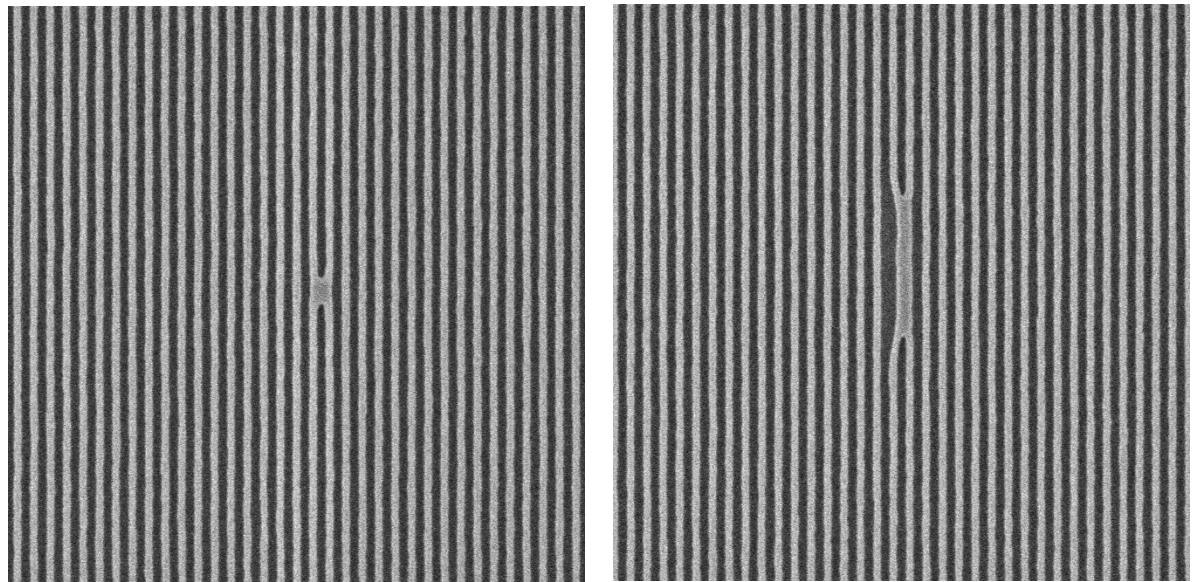
As we scale from node-to-node, device dimensions become smaller and smaller, and this brings unprecedented challenges to optical inspection as well as for e-beam inspection. Recently, e-beam based inspection has become more and more pertinent for extremely small defect detections. The graph below shows the inspection space w.r.t the available tools. For inspection e-beam is often more sensitive when compared to optical but classification remains a challenge for both methods. Also, defect location accuracy is better for e-beam based tools which are often linked to design databases. Even though resolution and location accuracy improve greatly with e-beam tools, absence of robust classification algorithms often lead to increased engineering time (as engineers manually classify

defects). We notice that even on commercially available software, classification is not robust. This has forced us to look for alternative methods/algorithms for more robust defect classification.



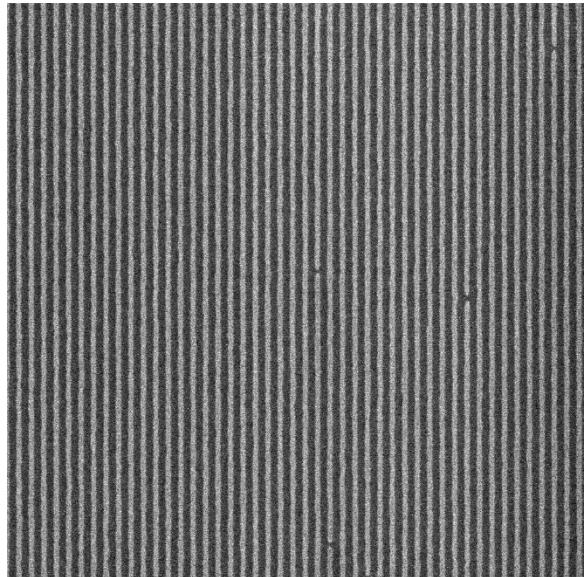
**Figure 4.1.** The Defect Inspection and Review Space.

Fig. 4.1 Shows the defect inspection and review space. E-beam inspection tools cover a multitude of different applications. While there have been some publications in the past in semiconductor manufacturing for BEOL applications [57]., there are not many for EUV defects, especially for bridges and collapses which are not trivial and where many commercial software often fails.

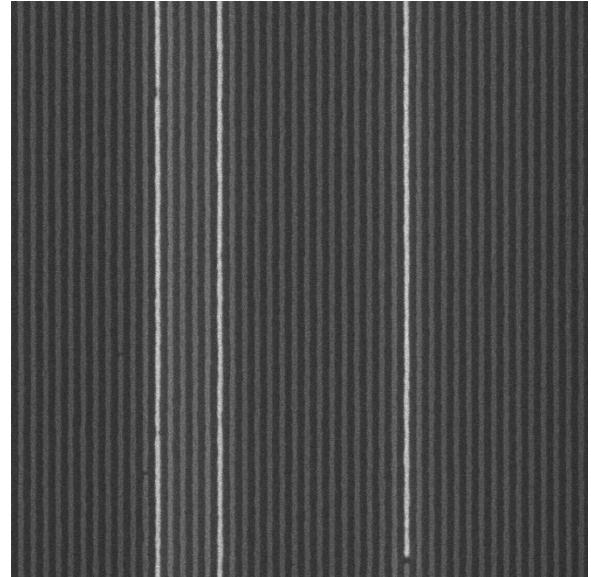


(a)

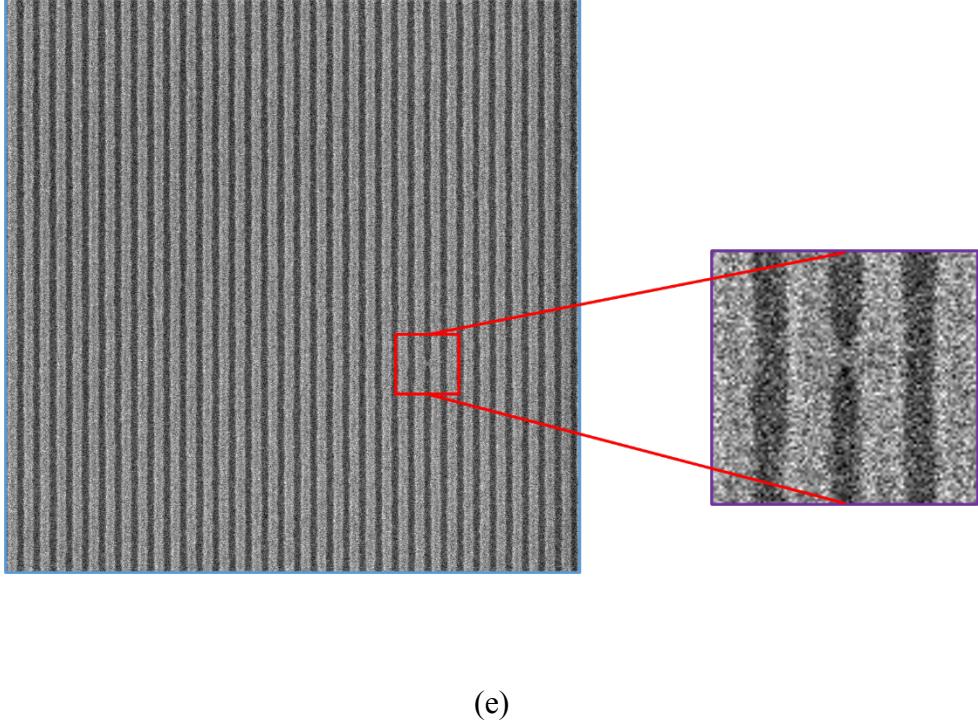
(b)



(c)



(d)



(e)

**Figure 4.2.** Typical Defects: (a) Bridge, (b) Line-Collapse, (c) Gap & Prob-Gap, and Challenging Defects: (d) Gap & Prob-Gap in presence of contrast change (e) Micro-bridges. Reprinted with permission from [84].

These defects may range from being critical failures to wafer-yield limiters. Fig. 4.2 shows SEM images with examples of different defect categories generally encountered in aggressive pitches. Fig. 4.2 (a), (b) and (c) are examples of Line-Space (L/S) patterns with defect type Bridge, Line-Collapse and Broken-line/Gap category, respectively. We depicted more challenging defect scenarios in Fig. 4.2 (d), as an example of Broken-line/Gap and probable-gap (partial feature missing) defect types in presence of contrast/intensity change (image intensity sometimes vary strongly from one line in the SEM image to another line due to different charging, when a line is broken, somewhere, not necessarily in the FOV) as well as in Fig. 4.2 (e), presence of random micro-bridges with variable degrees of pixel-level defect. The goal of this work is to show how deep learning (DL) based algorithms can be used for more robust classification of different defects during wafer processing after an optimal focus/dose is selected. In a previous work [33] we have already shown the benefit of using such techniques to help in drawing process

windows automatically from FEM wafers. In this chapter, we go a step further and show how such DL methods can classify the tougher bridge/collapse defects together with other process defects. In summary, there are five contributions in this research:

- A novel ensemble deep-learning model is proposed to solve challenging defect detection problems in SEM images. Our goal is to accurately classify, detect and localize different defect categories for aggressive pitches and thin resists (High NA applications). We have trained RetinaNet models using different ResNet [58], VGGNet [59], SSD\_MobileNet\_v1 [60, 61] architectures as backbone and proposed a preference-based ensemble strategy to combine the output predictions from different models and achieve better performance on classification and detection of different defects. The proposed ensemble method demonstrates improvement of the average precision metric (mAP) of the most difficult defect classes.
- Defect inspection in ADI SEM (After Develop Inspection) images is the most challenging task as different sources of noise [62] generally shadow the detailed device feature information. This often leads to false defect detections and erroneous metrology. The challenge also lies for some resist profile, in differentiation and detection of minute bridges (micro), breaks (zones of probable breaks) and resist footing from these noisy SEM images. Therefore, we have applied an unsupervised machine learning strategy to denoise [56] the SEM images aiming to optimize the effect of stochastic noise on structured pixels and therefore, to remove the False-Positive defects (FP) for better metrology and enhanced defect inspection. We have repeated the defect inspection step with the same trained model parameters and have performed a comparative analysis for “robustness” & “accuracy” metric for both noisy/denoised image pairs with different “detection confidence score”. We also have fine-tuned the proposed model by training with denoised images for the above-mentioned challenging defect classes.

- We have analysed and validated our proposed model performance against conventional tools/approaches. We have noticed that while using the conventional approach, various defects are not being flagged and we believe that this limitation is due to the “manual” selection of the detection threshold parameter. Furthermore, the detection scenario is influenced by the condition if the image is noisy or denoised. However, our proposed model demonstrates “stable” performance in detecting defects with better accuracy for both noisy or denoised images and replaces the manual trial-and-error based “threshold” selection method with automated “confidence score”. Once defects are correctly detected, different parameters (as length, width, area, additional feature vectors) about the defects can be output for better understanding the root cause of the defects. Thus, our proposed approach demonstrates its effectiveness both quantitatively and qualitatively.
- We built an ADCD framework using state-of-the-art Mask R-CNN (Mask Regional Convolution Neural Network) [81]. To speed up the development cycle, we used Detectron2 [82], instead of developing a Mask R-CNN model from very scratch. In parallel to predict the class and box offset, Mask R-CNN has a third branch to output a binary mask for each defect region-of-interest (ROI). Therefore, the proposed method enables to extract and calibrate each segmented mask and quantify the pixels that make up each mask, which in turn enables us to count each categorical defect instances as well as to calculate the surface area in terms of pixels.
- Finally, we built an UI (User-Interface) using Streamlit library [63] to deploy our proposed model as a web-based defect inspection app. This will enable different customers/vendors to run the application on their local servers/workstations on their own tool data. This UI will enable the users to upload a dataset of SEM/EDR/Review-SEM images, to select and run the inference model on the dataset, to visualize the prediction

performance locally and finally to segregate and save the images in different folders according to their defect categorical classes in local machines.

## 4.2. Related Work

We briefly discussed existing research approaches and methodologies in the context of machine learning based defect inspection. Our search criteria are only limited to semiconductor process domain. Convolutional Neural Network or simply ConvNet gained popularity in the domain of computer vision applications following Yann LeCun's first introduced LeNet architecture [64], aimed to recognize handwritten digits. Since then, researchers have experimented with more complex architecture variants of CNN, as AlexNet [24], GoogleNet [65], VGGNet [59], ResNet [58], RetinaNet [66] etc., to correlate depth of architectures, especially convolutional blocks with model accuracy. Recurrent usage of these deep learning models can also be observed for robust defect inspection strategy in every single process step of any real-world production or manufacturing pipeline. Semiconductor industries are of no exception to this. M. Sharifzadeh et. al. [67] has investigated detection and classification of four steel surface defect categories as hole, scratch, coil break and rust using conventional image processing algorithms. Authors reported four different accuracy metrics for detecting above four popular kind of steel defects. The drawback of this proposed method is the “trial and error selection method” of the high-performance method among several tested image processing algorithms. Four most common operations for defect detection as (1) thresholding, (2) noise removal, (3) edge detection and (4) segmentation do not survive the requirements of the semiconductor industry at this advanced node era. S. Kim et. al. [68] proposed a novel defect detection approach using component tree representations of SEM images. They proposed a modified version of the original framework to better identify defects. The algorithm is based on the topographic map representation of the SEM images and therefore built a component tree in

quasi-linear time with respect to attributes such as area, height, volume, and stiffness etc. These attributes proved to be essential to define and detect meaningful defect regions. The authors proposed two versions of the framework, one to detect defects of uniform size and shape and another to tackle more difficult cases with variable size and shape. The drawback of this methodology is that the performance evaluation is only qualitative and not quantitative. There exists no standard dataset or standard evaluation protocol to carry out fair comparative analysis of this proposed approach. Ravi Bonam et. al. [69] has studied the effect of defect sizes and their impact on EUV lithography. It is observed that optimal value of manually tuned process parameters depends on defect types when technology node scale is of few nanometers. Also, the capture efficiency of the manual technique is directly proportional to the nuisance rate. J. Wang et. al. [70] proposed “AdaBalGAN” model (adaptive balancing generative adversarial network) to tackle misidentification problem of defective pattern recognition from of wafer maps due to imbalanced defective class data. An adaptive generative model is proposed to balance the number of samples of each defect category as per classification accuracy. Ji-Hee Lee et. al. [71] has proposed transfer learning mechanism in the scope of machine learning strategy to build a reliable defect detection method for patterned wafer images. Their idea is based on to cope with new categories of defect data, with an already trained model, in each update cycle with minimum data possible and with minimum engineering resources and time. They have shown how deep learning method is outperforming than traditional defect inspection algorithms both in accuracy and time. The drawback of this strategy is that it can only judge whether a defect exists or not but not able to classify the defect classes. B. Devika et. al. [72] proposed a CNN based deep learning model to identify wafer defect patterns. The authors have taken into consideration the Wafer Bin Map (WBMs) patterns as circle, cluster, scratch, and spot as since each defect pattern correlates with different fabrication

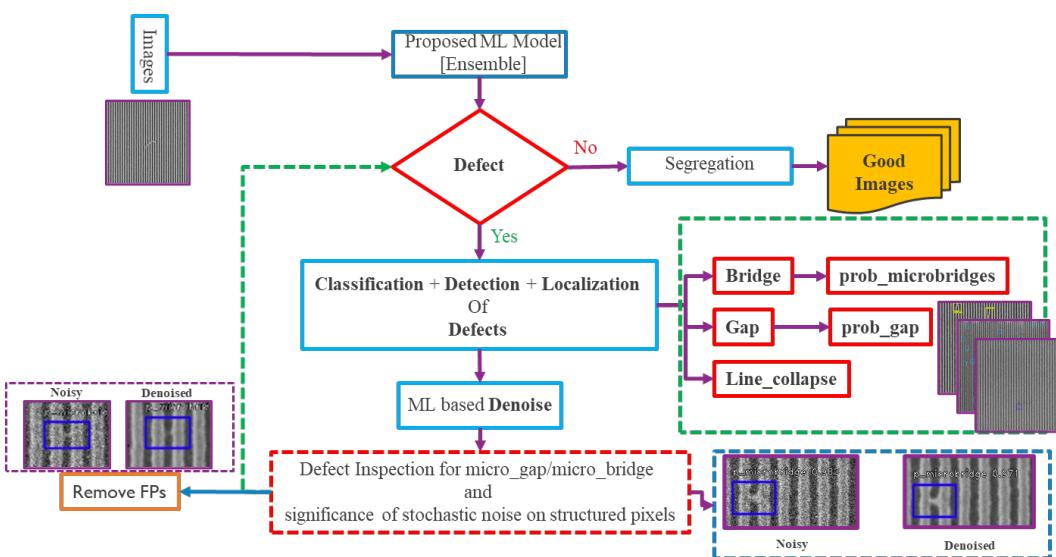
errors. An early and efficient ML based defect detection strategy leads to reduced wafer test time and an improved die yield. Jong-Chih Chien et. al. [73] has proposed two ways to use deep CNN architecture to classify semiconductor defect images. The defect classes were aimed as center, local, random and scrape. Their approaches were not tested on mixed defect types as well as the authors reported about occurrence of misclassification during validation phase which demands further fine-tuning of the model. Y. Yuan-Fu [74] proposed alternative machine learning techniques against automatic optical inspection (AOI) to visualize defect patterns and to identify root causes of die failures. The limitation of AOI method is that it still requires human expert intervention to judge the type of defect. In this paper, the authors proposed CNN and XGBoost techniques to retrieve wafer maps and to classify the defect patterns. They compared the classification performance of the proposed method against random decision forests (RF), support vector machine (SVM), adaptive boosting (Adaboost). Dhruv V. Patel et. al. [75] demonstrated the effectiveness of optimized deep learning models in identifying, localizing, and classifying different types of wafer defects with high degree of accuracy. They have generated high-resolution EB images of wafers patterned with different types of intentional defect categories and trained their CNN based models. They have also demonstrated the significance of CAM (Class Activation Maps) for localizing the defects. Joongsoo Kim et. al. [57] proposed a CNN based defect image classification model derived from Residual Network to classify defects specialized for TSV (Through-Silicon-Via) process. Image pre-processing has been performed before the model deployment to increase classification accuracy as well as to tackle size dependent defect classification issue.

We have carefully examined all the proposed methodologies, solutions addressed and most importantly the limitations as discussed by previous authors to formulate our proposed approach.

### 4.3. Proposed Method

Our proposed approach, based on RetinaNet, to detect different defects from SEM images and classify them according to their corresponding classes in aggressive pitches is presented. Our proposed ensemble model-based defect detection framework, which is illustrated in Fig. 4.3, consists of the RetinaNet based detector and the U-Net architecture based denoiser. The framework is trained and evaluated using imec datasets (both Post-Litho and Post-Etch resist Wafer dataset) and classifies, detects, and localizes the candidate defect types. The focus of this section is to briefly discuss the key modules of the defect detection network only. We have utilized U-Net architecture based deep learning denoiser following [56, 76] and therefore beyond the scope of this section. To the best of our knowledge, this framework is the first to apply a novel robust supervised deep learning training scheme to accurately classify as well as localize different defect types in SEM images. The key modules of the defect detection network are:

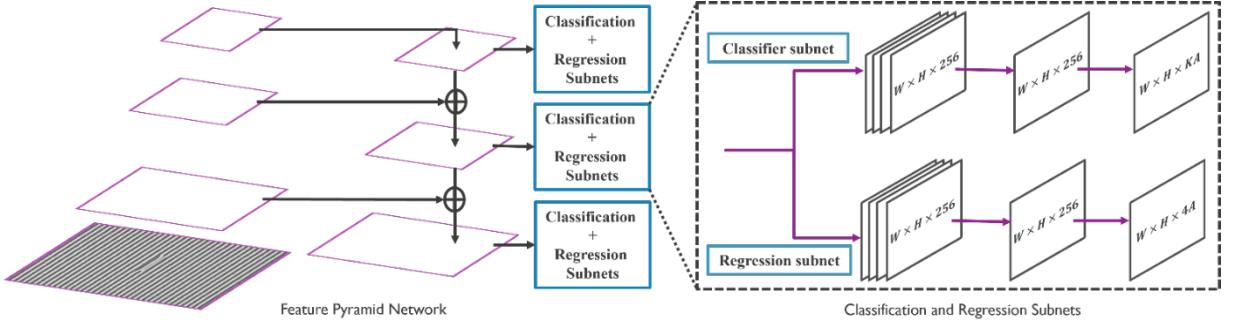
- RetinaNet defect detector architecture
- Deep feature extractor networks as backbone



**Figure 4.3.** Proposed ensemble model-based defect detection framework. Reprinted with permission from [84].

### 4.3. A. Overview of the RetinaNet architecture.

RetinaNet is popular



**Figure 4.4.** RetinaNet defect detector architecture. Reprinted with permission from [84].

one-stage object detection model which works well with dense objects and effectively handles the foreground-background class imbalance problem affecting the performance of other one-stage detector models. RetinaNet architecture consists of a Feature Pyramid Network (FPN) [77] built on top of a deep feature extractor network, followed by two subnetworks, one for object classification and the other for bounding box regression. RetinaNet defect detector architecture is illustrated in Fig. 4.4.

FPN takes one single resolution input image, subsamples it into multiple lower resolution images and outputs the feature maps at different scales, thus building a multi-scale feature pyramid representation. Thus, it enables detection of objects of varying sizes from different layers of the feature pyramid. FPN combines low resolution features with high resolution features via a top-down pathway which has lateral connections to layers from a bottom-up pathway. The bottom-up pathway generates a feature hierarchy using feature maps of different scales from the input image. The top-down pathway performs up-sampling on the

spatially coarser feature maps coming from the higher pyramid levels. The lateral connections are then used to merge the feature maps of same spatial size from both the paths, which gives semantically strong feature maps.

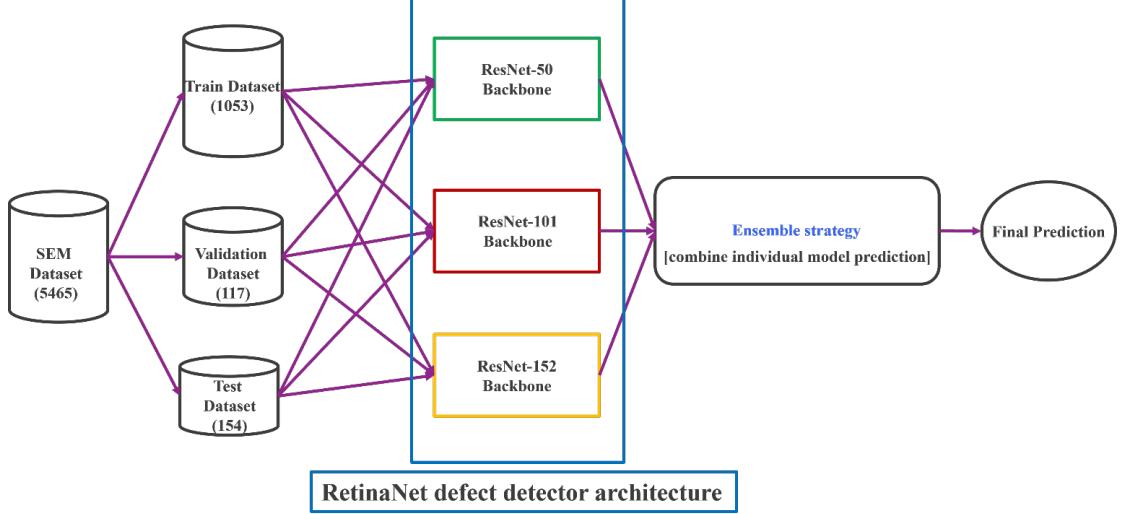
The classification and regression subnetworks are connected to every layer of the feature pyramid and are independent from each other. The classification subnetwork predicts the probability of an object's presence for every anchor box and object class. It consists of 4 fully convolutional layers [( $3 \times 3$ ) Conv layer with 256 filters and ReLU activation]. It follows another ( $3 \times 3$ )convolutional layer having  $K \times A$  filters where  $K$  is the number of classes, and  $A$  is the number of anchors ( $A = 9$  anchors covering 3 different aspect ratios and 3 different scales).

The regression subnetwork regresses the anchor boxes' offset against the ground truth object boxes. It is a class-agnostic regressor that does not know what class the objects belong to and uses fewer parameters. The structure is similar to the classification network, except it outputs four bounding box coordinates for every anchor box. Anchor boxes are assigned to ground truth boxes if the IOU between the boxes exceeds 0.5 and assigned to background if the IOU is in the range [0,0.4). The anchor boxes having IOU in the range [0.4,0.5) are ignored.

RetinaNet uses focal loss [66], which improves the prediction accuracy giving more importance to the hard samples during training and reducing the contribution of easy samples to the loss. It enhances the cross-entropy loss by introducing a weighting factor to offset the impact of class imbalance and a modulating factor to focus more on training the hard negatives and less on the easy examples.

#### **4.3. B. Deep Feature Extractor Networks as Backbone**

The proposed RetinaNet defect



**Figure 4.5.** Deep feature extractor networks as backbone. Reprinted with permission from [84].

detector framework is an ensemble architecture based on a selective permutation of backbones as **ResNet50**, **ResNet101** and **ResNet152** as shown in Fig. 4.5. Table. 4.1 describes custom variants of ResNet architectures with multiple convolutional layers with skip connections across them for feature extraction and fully connected layers for predicting different defect category probabilities. We have taken the affirmative ensemble [78] of the predictions from the 3 ResNet models with preference to the models showing better performance on the test dataset. So, we consider all the predictions from the first model and then we add those predictions from the second-best model which are not overlapping with the first model predictions. We use an IOU threshold of 0.5 to consider the boxes as overlapping. In this way, we add the non-overlapping predictions from the third-best model. This ensemble strategy ensures that all the predictions from the 3 models are taken, and this improves the accuracy of the test dataset.

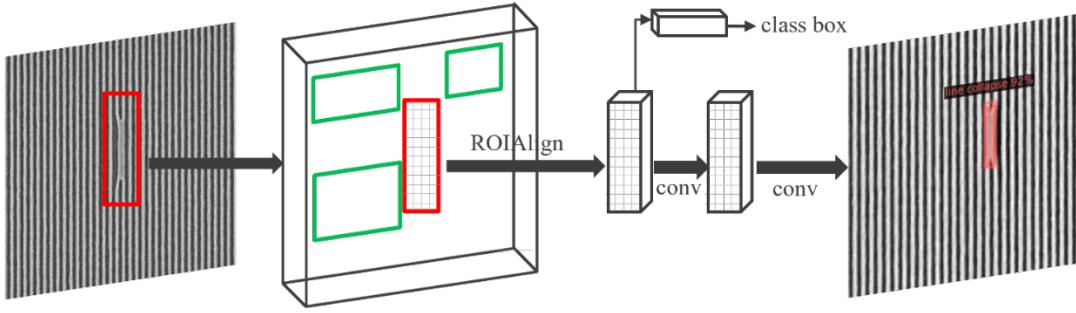
**Table 4.1.** ResNet50, ResNet101, and ResNet152 backbone architecture. Reprinted with permission from [84].

Layer Name	ResNet-50 Backbone	ResNet-101 Backbone	ResNet-152 Backbone
<b>Conv1</b>	<b><math>7 \times 7, 64, \text{stride } 2</math></b>		
	<b><math>3 \times 3, \text{max pool, stride } 2</math></b>		
<b>Conv2_x</b>	$\begin{bmatrix} 1 \times 1 & 64 \\ 3 \times 3 & 64 \\ 1 \times 1 & 256 \\ \times 3 \end{bmatrix}$	$\begin{bmatrix} 1 \times 1 & 64 \\ 3 \times 3 & 64 \\ 1 \times 1 & 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1 & 64 \\ 3 \times 3 & 64 \\ 1 \times 1 & 256 \end{bmatrix} \times 3$
<b>Conv3_x</b>	$\begin{bmatrix} 1 \times 1 & 128 \\ 3 \times 3 & 128 \\ 1 \times 1 & 512 \\ \times 4 \end{bmatrix}$	$\begin{bmatrix} 1 \times 1 & 128 \\ 3 \times 3 & 128 \\ 1 \times 1 & 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1 & 128 \\ 3 \times 3 & 128 \\ 1 \times 1 & 512 \end{bmatrix} \times 8$
<b>Conv4_x</b>	$\begin{bmatrix} 1 \times 1 & 256 \\ 3 \times 3 & 256 \\ 1 \times 1 & 1024 \\ \times 6 \end{bmatrix}$	$\begin{bmatrix} 1 \times 1 & 256 \\ 3 \times 3 & 256 \\ 1 \times 1 & 1024 \\ \times 23 \end{bmatrix}$	$\begin{bmatrix} 1 \times 1 & 256 \\ 3 \times 3 & 256 \\ 1 \times 1 & 1024 \\ \times 36 \end{bmatrix}$
<b>Conv5_x</b>	$\begin{bmatrix} 1 \times 1 & 512 \\ 3 \times 3 & 512 \\ 1 \times 1 & 2048 \\ \times 3 \end{bmatrix}$	$\begin{bmatrix} 1 \times 1 & 512 \\ 3 \times 3 & 512 \\ 1 \times 1 & 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1 & 512 \\ 3 \times 3 & 512 \\ 1 \times 1 & 2048 \\ \times 3 \end{bmatrix}$
<b>FC</b>	<b>Average pool, 1000-d fc, softmax</b>		

The goal of our proposed ensemble model-based defect detection framework is based on two significant steps. In the first step, train a RetinaNet defect detector architecture as discussed above to accurately classify as well as localize different defect types in SEM images such as bridge, line\_collapse, gap, micro-bridges, and micro-gaps, respectively. In the second step, denoise those SEM images with challenging defects such as micro-bridges and micro-gaps to optimize the effect of stochastic noise on structured pixels. We then reiterate the defect detection step to remove the False-Positive defects (FP) towards better metrology and enhanced defect inspection.

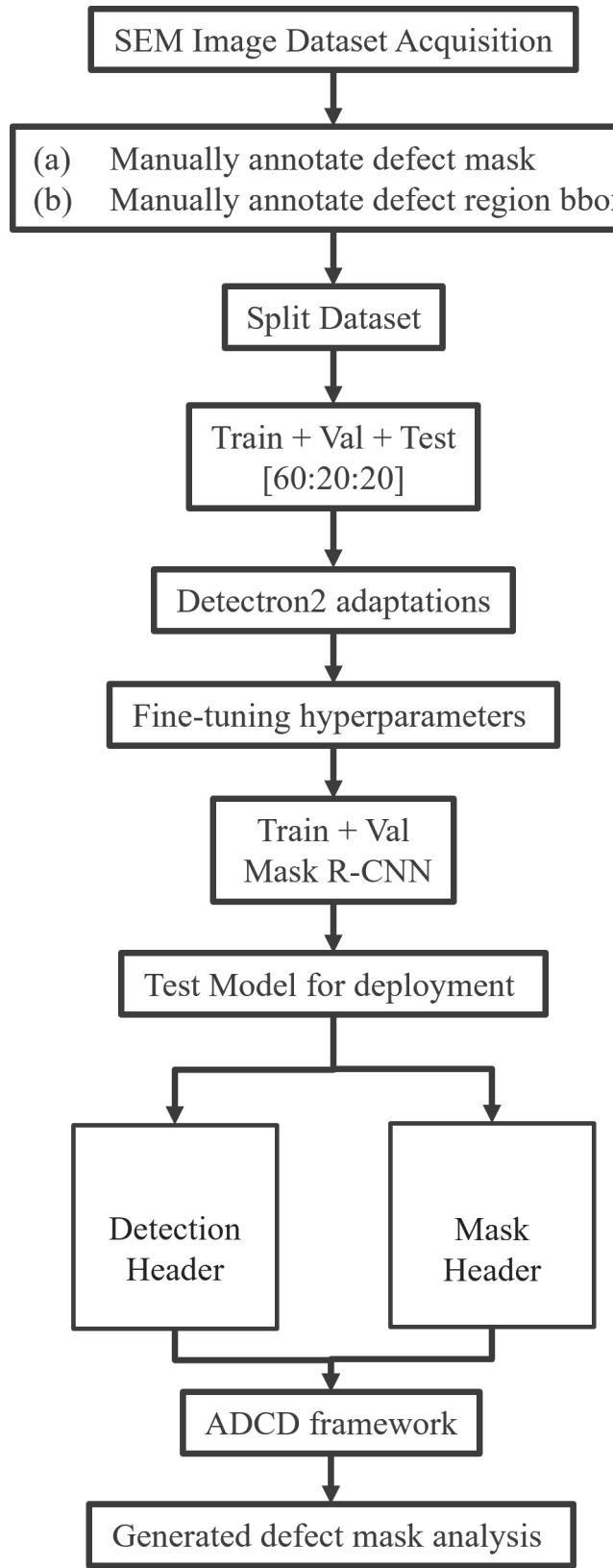
### 4.3. C. Mask R-CNN

The proposed methodology is based on Mask R-CNN, introduced by Facebook AI Research (FAIR), which combines the Faster R-CNN and FCN



**Figure 4.6.** Mask R-CNN framework for defect instance segmentation

(Fully Connected Network) to get additional mask output other than the class and box outputs. Mask R-CNN architecture adopts (1) ROI align operation, replacing ROI-pooling in Faster R-CNN, to allow to create precise instance segmentation masks. This is our goal here for extending previously proposed RetinaNet defect detector framework for defect classification and detection, (2) adds a network head as FCNN to produce the required defect instance segmentations, and (3) independent mask and detection header as shown in Fig. 4.6. The proposed model is implemented using backbones as (a) faster+ResNet50+FPN, and (b) faster+ResNet101+FPN. Fig. 4.7 depicts the steps involved for training Mask R-CNN model for segmenting a defect instance and generating a binary mask for each ROI.



**Figure 4.7.** Flowchart of Mask R-CNN based ADCCD framework

#### 4.4 Implementation

We implemented the ensemble model-based defect detection framework using Keras [27] and Tensorflow [28]. For training, we have used the Keras implementation in [83]. The Anaconda version was 4.9.2. Our model has been trained and evaluated on Lambda TensorBook with NVIDIA RTX 2080 MAX-Q GPU.

#### 4.4 A. Datasets

The proposed ensemble model (Classifier + Detector) is trained and evaluated on both Post-Litho and Post-Etch P32 (Pitch 32 nm) resist wafer dataset. The dataset consists of a total of 5,465 raw SEM images of (1024 × 1024) pixels in TIFF format with stochastic defects such as bridge, line-collapse, gaps/line-breaks, micro/nano-bridges, and probable nano-gaps as well as clean images without any such defects. The representative defect class images from this dataset are already shown in Fig. 4.2 (a) – (e). We have manually labeled 1170 SEM images (1053 training images, 117 validation images) images using LabelImg [79] graphical image annotation tool. The defect labeling strategy comprises diverse defect representative and challenging condition instances and as per naming convention in Table. 4.2. The dataset was divided into a training set, a validation set, and a test set as shown in Table. 4.3. We have a total of 2529 defect instances of these 5 different defect classes for training and 337 instances for validation. To comply with training criteria, we converted all images with “.tiff” format into “.jpg” format. We also implemented different data-augmentation techniques (as rotation, translation, shearing, scaling, flipping along X-axis and Y-axis, contrast, brightness, hue, and saturation) to balance/increase the diversity of training dataset defect patterns. We did not consider using any digital twins or synthetic datasets as cited in some previous citations as that does not solve the purpose of tackling

real FAB originated stochastic defectivity scenario. We have also excluded any fabricated dataset patterned with intentionally placed or programmed defect types.

**Table 4.2.** Defect class labeling convention. Reprinted with permission from [84].

DEFECT CATEGORY	LABELLED AS
BRIDGE	<b>bridge</b>
LINE-COLLAPSE	<b>line_collapse</b>
GAP/LINE-BREAKS	<b>gap</b>
MICRO/NANO-BRIDGE	<b>microbridge</b>
PROBABLE NANO-GAP	<b>p_gap</b>

**Table 4.3.** Data distribution of defect SEM images. Reprinted with permission from [84].

Class Name	Train (1053 images)	Val (117 images)	Test (154 images)
<b>gap</b>	1046	156	174
<b>p_gap</b>	315	49	54
<b>microbridge</b>	380	47	78
<b>bridge</b>	238	19	17
<b>line_collapse</b>	550	66	76
<b>Total Instances</b>	2529	337	399

#### 4.4 B. Evaluation Criteria

We have considered Intersection over Union (IoU) [80] between the ground truth bounding box and the predicted bounding box  $\geq 0.5$ . The “defect detection confidence score” metric is taken as 0.5. The proposed ensemble model-based defect detector overall performance is evaluated against mAP as Mean Average Precision, where mAP is calculated using the weighted average of precisions among all defect classes. AP or average precision provides the detection precision for one specific defect class. We have also considered the speed of detection per image (average-inference-time in milliseconds). We have taken the affirmative ensemble [78] of the predictions from top  $k$  backbones with preference to the models showing better performance on the test dataset. So, we consider all the predictions from the first model and then we add those predictions from the second-best model which

are not overlapping with the first model predictions. We use an IOU threshold of 0.5 to consider the boxes as overlapping. In this way, we add the non-overlapping predictions from the third-best model and so on up to  $k$  models. This ensemble strategy ensures that all the predictions from the top  $k$  backbones are taken, and this improves the accuracy of the test dataset. The improvement is noticeable for the most difficult defect category **p\_gap** where the ensemble precision exceeds the individual model precisions.

#### 4.4 C. Training

We have first trained experimentally the different individual backbone architectures (ResNet50, ResNet101, ResNet152, SSD\_MobileNet\_v1, SeResNet34, Vgg19 and Vgg16) on our SEM image dataset independently. For the proposed experiments, we have selected training parameters and hyperparameters as: 40 epochs, batch-size of 1, initial learning rate at 0.00001, learning rate reduction by a factor of 0.1 if learning rate plateaus, and optimizer as ADAM [31]. Table 4.4 provides the comparison analysis for defect detection accuracies obtained per defect class as well as mAP on test images for the above experimental backbones with score-threshold 0.5. We have selected top three ResNet architectures with 78.7%, 77.5% and 78.8% mean average precision (mAP) respectively and SSD\_MobileNet\_v1 with 92.5% average precision (AP) for line\_collapse defect as our final candidate backbones for proposed ensemble model framework while discarding the others for worst average precision accuracy per defect class. The focal loss strategy is implemented for the proposed ensemble model with a weighting factor of  $\alpha = 0.25$  and focusing parameter  $\gamma = 2.0$ . This helps tackle the class imbalance problem and learn from challenging defect instances. Table 4.5 presents Test and Validation detection accuracies of top 3 ResNet architecture backbones per defect class along with average inference time in seconds. The proposed RetinaNet framework is an ensemble architecture based on a selective permutation of backbones as ResNet50→ResNet152→ResNet101. The selection

criterion as described in **4.4 B**, is justified to propose a preference-based ensemble strategy to combine the output predictions from different models and achieve better performance on classification and detection of different defects. As presented in Table **4.6**, our proposed ensemble approach achieves better results with overall mAP of 81.6% than the results obtained by 3 top individual backbones separately as shown in Table **4.5**. There is further scope of improvement of overall mAP metric again considering ensembling of SSD\_MobileNet\_v1 architecture as a backbone with 92.5% average precision (AP) for line\_collapse defect as a future work.

**Table 4.4.** Implementation results when experimenting with different backbone architectures. Reprinted with permission from [84].

Class Name	ResNet <b>50</b>	ResNet <b>101</b>	ResNet <b>152</b>	MobileNet224 <b>_1.0</b>	SeResNe t34	Vgg <b>19</b>	Vgg <b>16</b>
gap_AP	0.954	<b>0.968</b>	0.963	0.462	0.034	0.95 8	0.93 3
p_gap_AP	<b>0.432</b>	0.291	0.376	0.00	0.00	0.11 8	0.23 5
bridge_AP	<b>0.872</b>	0.811	0.844	0.723	0.717	0.73 2	0.78 6
microbridge_AP	0.603	0.633	0.669	0.104	0.003	0.7	<b>0.71</b> <b>5</b>
line_collapse_AP	0.828	0.816	0.789	<b>0.925</b>	0.925	0.79 9	0.78 8
<b>mAP</b>	<b>0.787</b>	<b>0.775</b>	<b>0.788</b>	0.429	0.222	0.75 4	0.76 2

**Table 4.5.** Test/Validation accuracy of top 3 ResNet architecture backbones. Reprinted with permission from [84].

Score_threshold: <b>0.5</b>	Test			Validation		
Class Name	ResNet <b>50</b>	ResNet1 <b>01</b>	ResNet1 <b>52</b>	ResNet <b>50</b>	ResNet1 <b>01</b>	ResNet1 <b>52</b>
gap_AP	0.954	<b>0.968</b>	0.963	<b>0.969</b>	0.963	0.947
p_gap_AP	<b>0.432</b>	0.291	0.376	<b>0.346</b>	0.232	0.28
bridge_AP	<b>0.872</b>	0.811	0.844	0.927	0.894	<b>0.947</b>
microbridge_AP	0.603	0.633	<b>0.669</b>	0.738	<b>0.792</b>	0.786
line_collapse_AP	<b>0.828</b>	0.816	0.789	<b>0.909</b>	0.864	0.864
<b>mAP</b>	0.787	0.775	<b>0.788</b>	<b>0.832</b>	0.809	0.811
average_inference_time (Secs)	0.0769	<b>0.0656</b>	0.0782	--	--	--

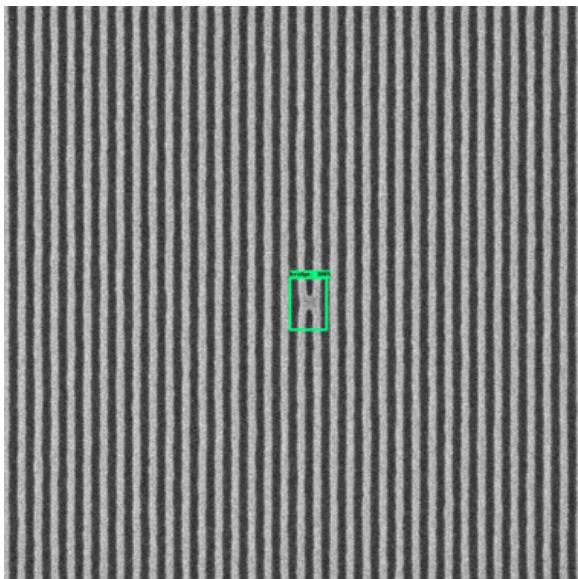
**Table 4.6.** Overall test accuracy of proposed RetinaNet [ensemble\_ResNet] framework. Reprinted with permission from [84].

Proposed Model	Ensemble_ResNet [ResNet50→ResNet152→ResNet101]
gap_AP	0.959
p_gap_AP	0.52
bridge_AP	0.867
microbridge_AP	0.675
lineCollapse_AP	0.828
mAP	<b>0.816</b>

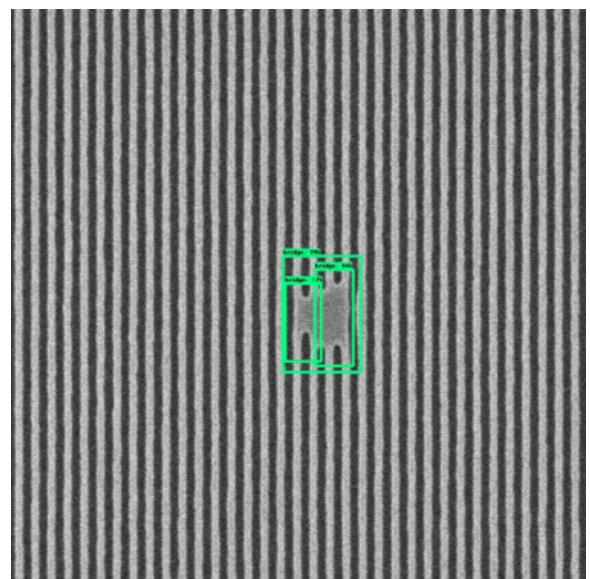
## 4.5. Experimental Results

### 4.5 A. Ensemble\_ResNets Defect Detection Performance:

Examples of typical defect classification and detection results are shown in [Fig. 4.8-Fig. 4.10] (a) (b) as single or multiple defect instances of bridge, line-collapse and gap/line-break.

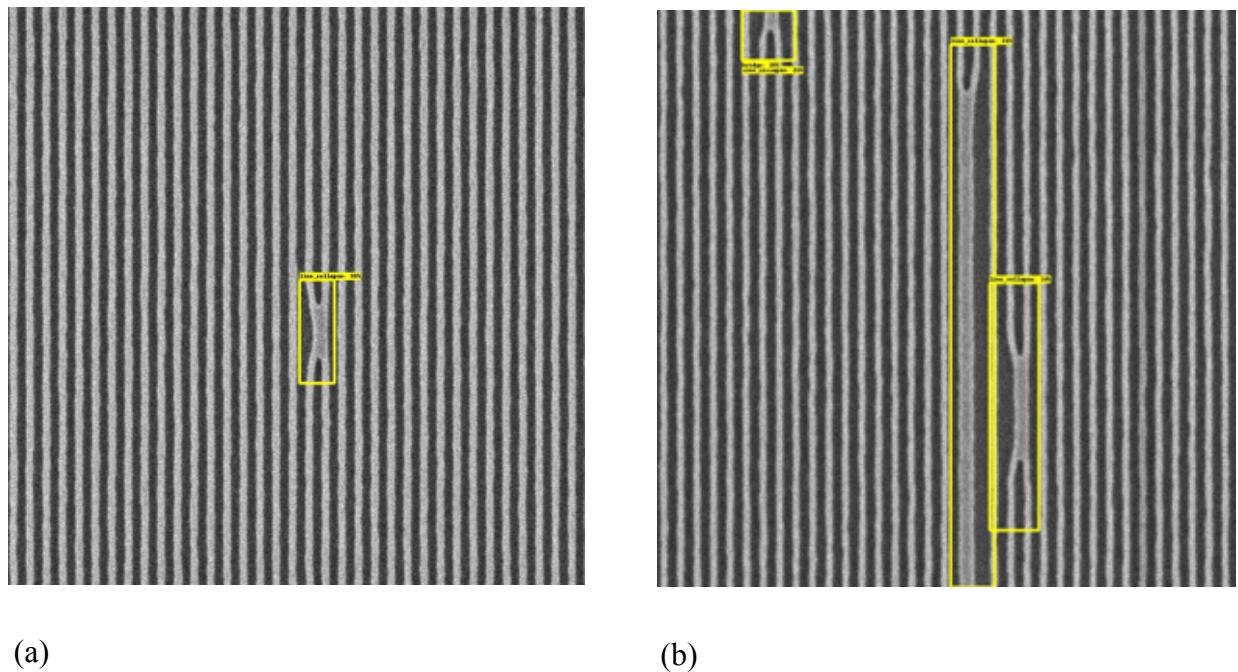


(a)

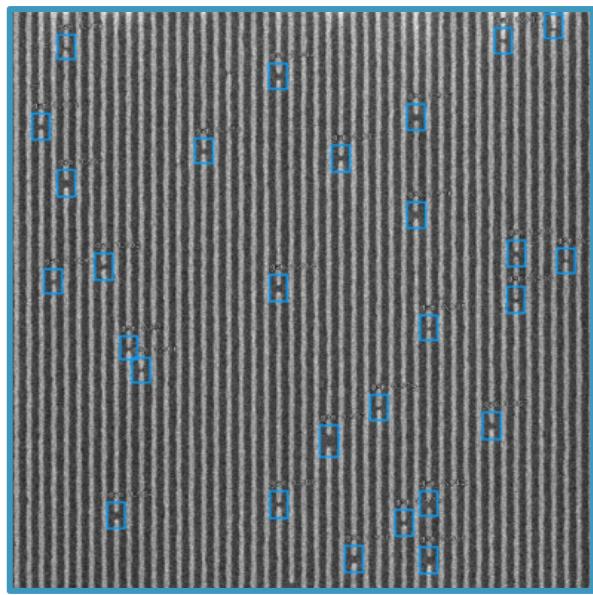


(b)

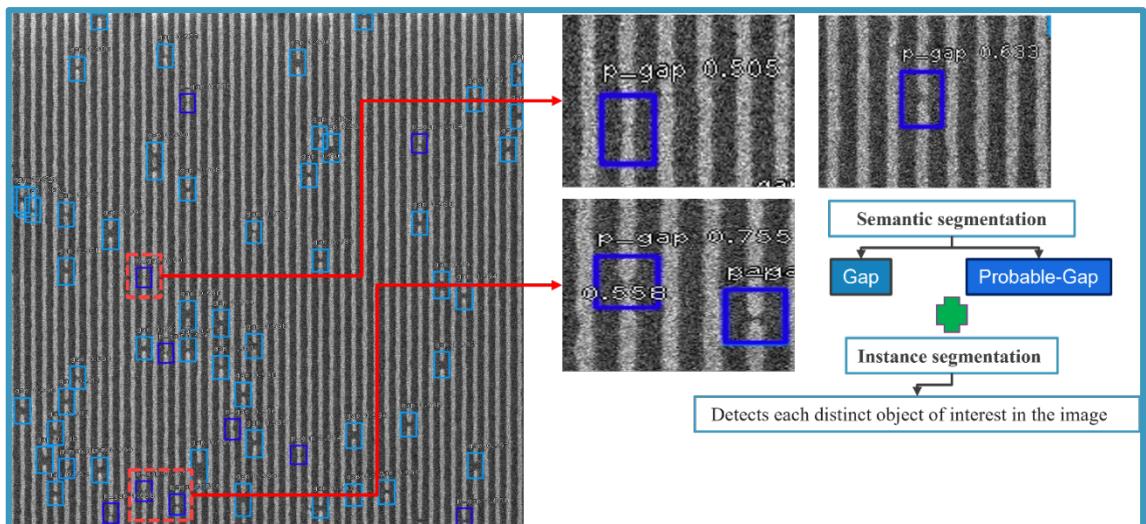
**Figure 4.8.** BRIDGE detection results with confidence score.(a) Single Bridge, (b) Multiple Bridges. Reprinted with permission from [84].



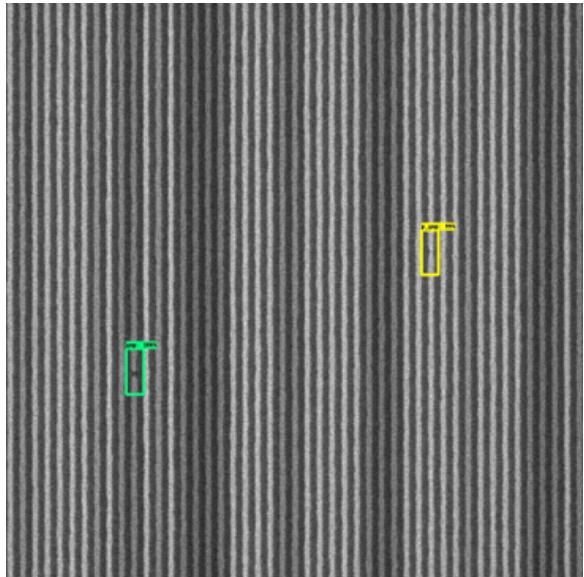
**Figure 4.9.** LINE-COLLAPSE detection results with confidence score.(a) Single Line-Collapse, (b) Multiple Line-Collapses. Reprinted with permission from [84].



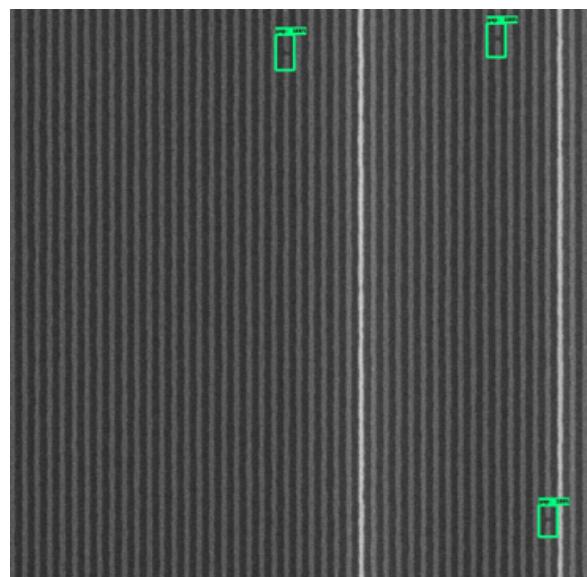
**Figure 4.10.** GAP/BREAK detection results with confidence score. Multiple Line-breaks. Reprinted with permission from [84].



**Figure 4.11.** Detection results of more challenging Probable NANO-GAP separately in presence of GAP/BREAK. Model shows robustness in detecting relatively few probable Nano-Gap defects in presence of frequent Gap defectivity. Reprinted with permission from [84].

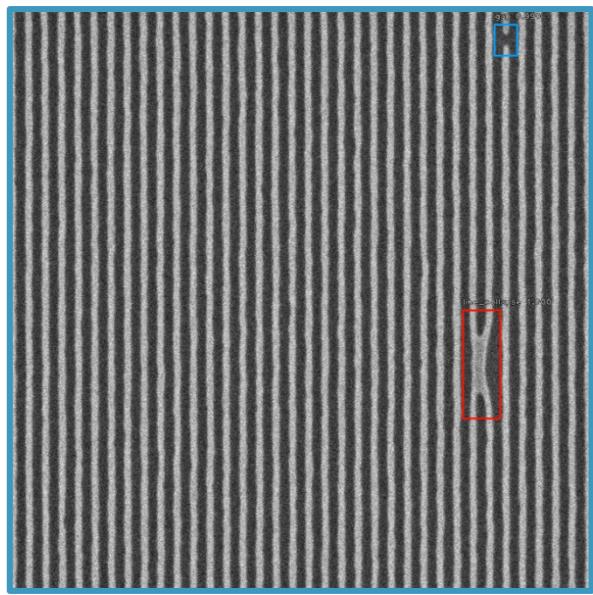


(a)

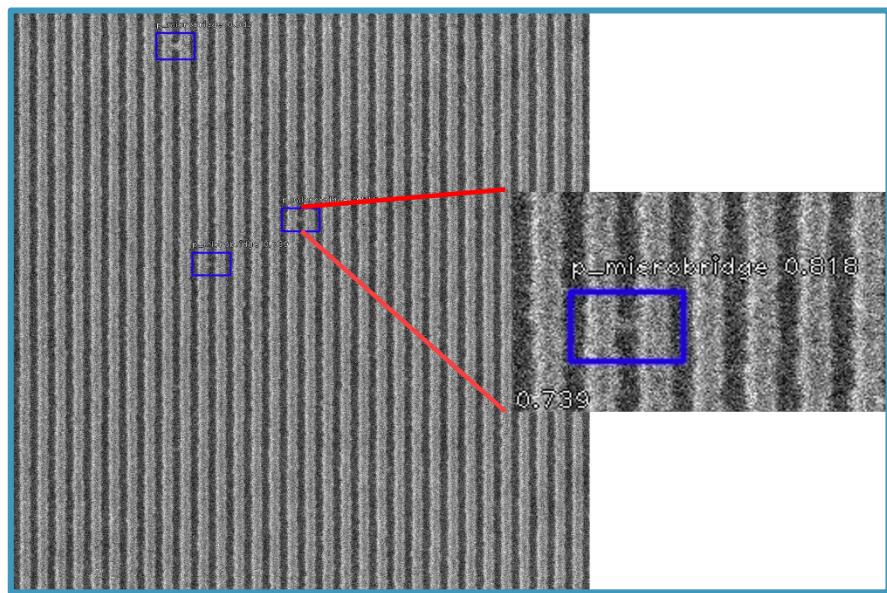


(b)

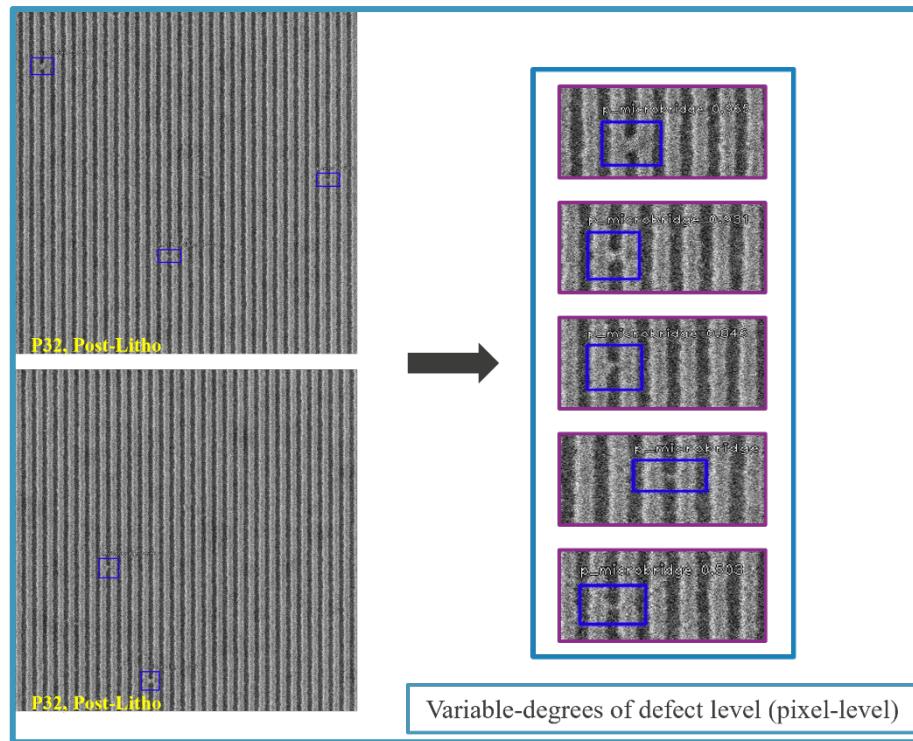
**Figure 4.12. (a) (b).** Detection of NANO-GAPs and Probable NANO-GAPs in presence of contrast change scenario. Contrast change does not affect defect detection performance of proposed ML model in comparison to conventional approach. Reprinted with permission from [84].



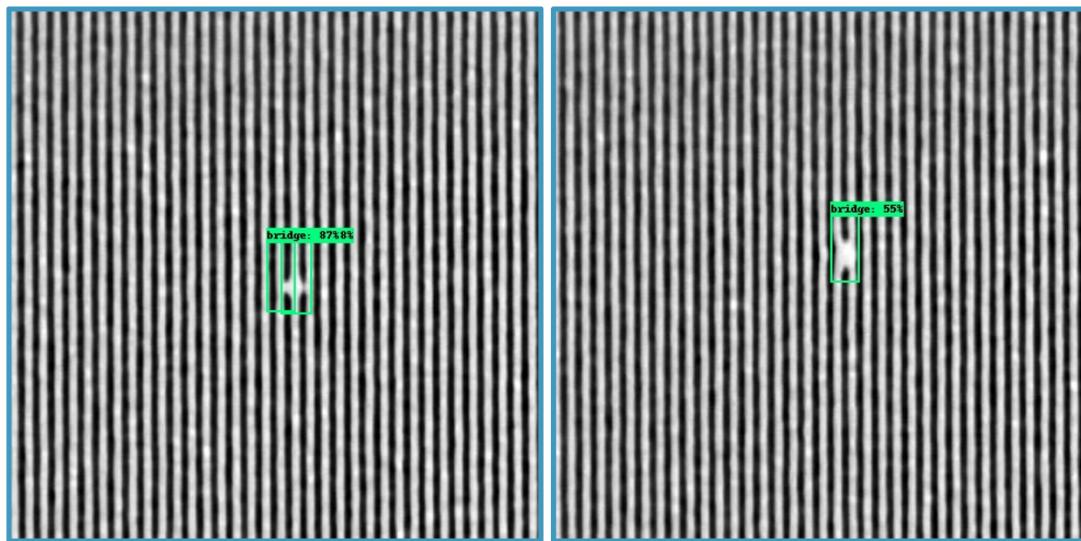
**Figure 4.13.** Mixed defect detection results with confidence score. (a) Single Line-Collapse, (b) Single Gap/Break



**Figure 4.14.** Detection results of more challenging NANO-BRIDGE/MICRO-BRIDGE defectivity. Reprinted with permission from [84].



**Figure 4.15.** Detection results of more challenging NANO-BRIDGE/MICRO-BRIDGE defectivity on new TEST dataset. Model demonstrates robustness in detecting variable degrees of pixel-level micro-bridge defectivity. Reprinted with permission from [84].



**Figure 4.16.** Defect detection on Review-SEM images.

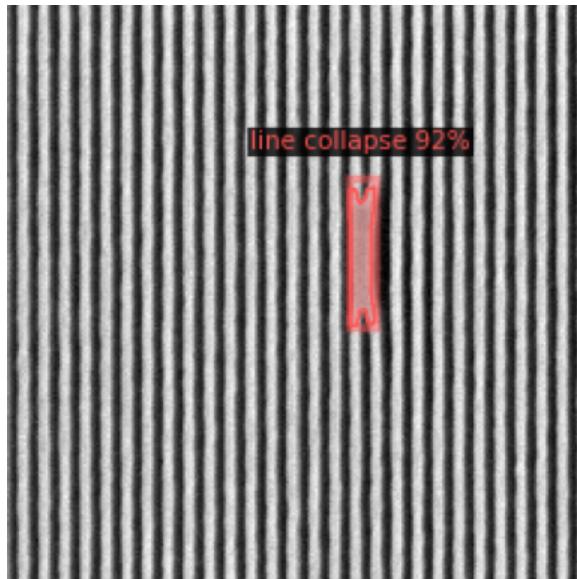
Fig. 4.11 shows robustness of our proposed model in detecting relatively few more challenging probable nano-gap defects in presence of frequent gap defectivity. It emerged

as a very challenging scenario for conventional approaches or tools Ref. 5 to differentiate between these two marginal defect categories. However, our proposed model demonstrates (1) semantic segmentation between two distinct defect classes as (a) gap/line-break and (b) probable nano-gap and (2) instance segmentation as detection of each distinct defect of interest under these two defect classes in the same image. Fig. 4.12(a)(b) illustrates detection of nano-gaps and probable nano-gaps in presence of contrast change scenario. We can see contrast change does not affect defect detection performance of our proposed model against conventional approach [16]. Fig. 4.13 depicts defect detection performance when mixed defect categories are present in a same image. Figs. 4.14, 4.15 shows detection results of more challenging nano-bridge/micro-bridge defectivity on new test image dataset. To validate the proposed model performance and robustness, we have run the defect detection inference model on previously unseen SEM image dataset with different resist family. Composition of a resist is a significant variable that have an impact on the number of stochastic defects as well as its pixelsize like microbridge and probable nano-gap defects, respectively. Our proposed deep learning-based model demonstrates robustness in detecting variable degrees of pixel-level micro-bridge defectivity (detect individual microbridges regardless their extent). Fig. 4.16 shows defect detection on review-SEM images. Proposed model, as trained with CD-SEM images, shows robust defect detection capability on Review-SEM images (thus different test distribution). Hence, we demonstrated the ability of our proposed framework to generalize over different SEM applications as well as an assist tool for better defect inspection in the production lines of semiconductor industry. Our proposed ensemble model-based defect detection framework achieves the detection precision (AP) of **95.9%** for gap, **86.7%** for bridge, **82.8%** for line\_collapse, **67.5%** for microbridge, and **52.0%** for probable nano-gap defectivity, respectively. However, we believe there is further scope of improvement for average

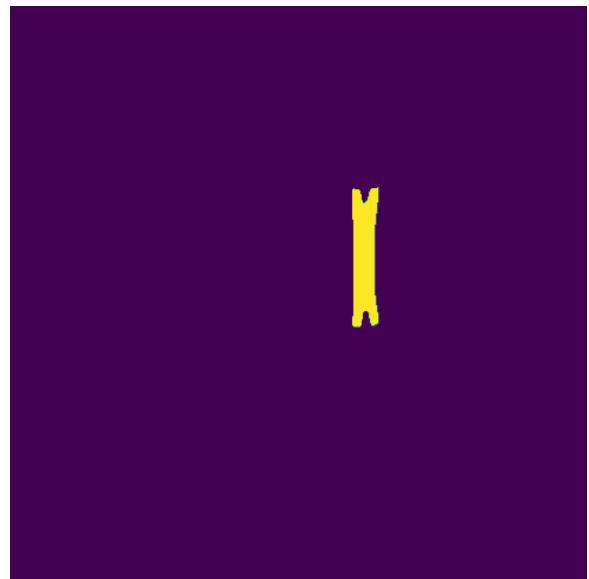
precision for specific classes like microbridge and probable nano-gap, thus overall mAP of the proposed framework can also be improved. This will be considered as our next step of this research.

#### 4.5 B. Mask R-CNN Defect Detection Performance

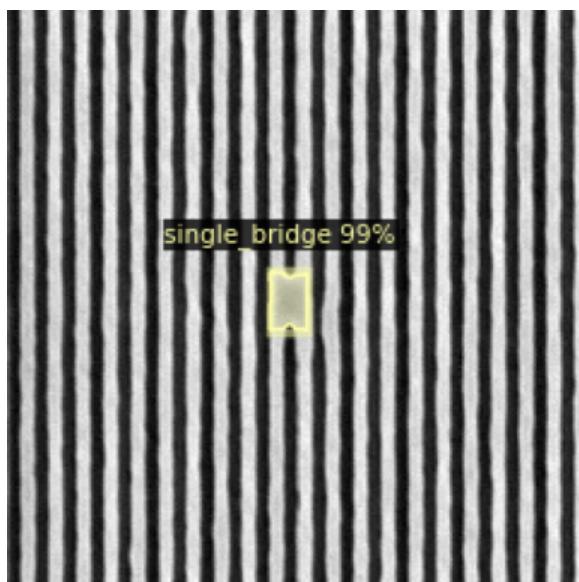
Fig. 4.17 (a)-(h) shows examples of defect detection on test images.



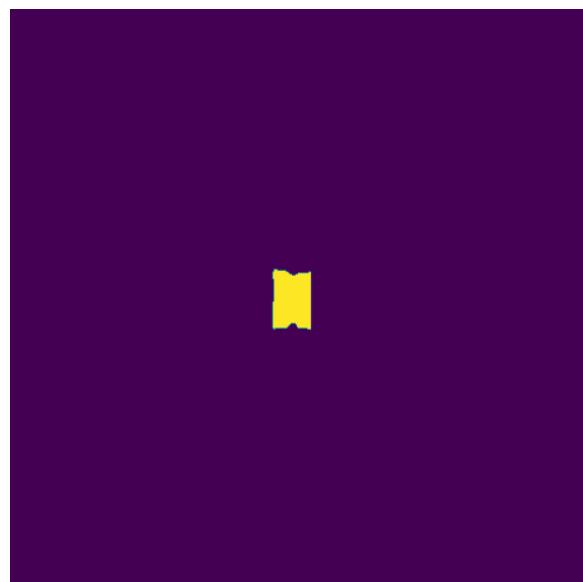
(a)



(b)



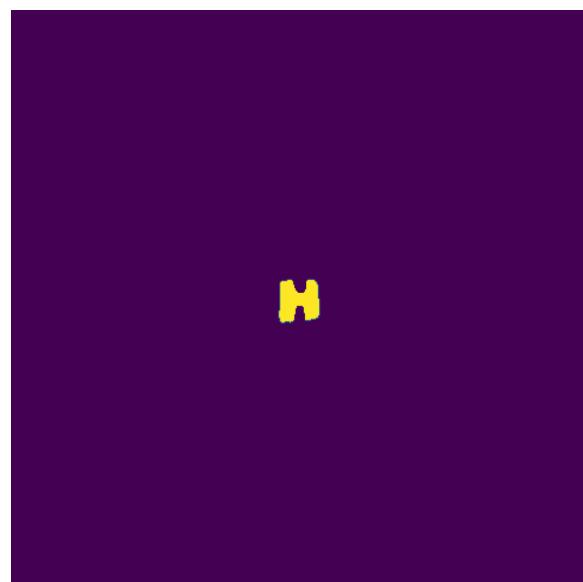
(c)



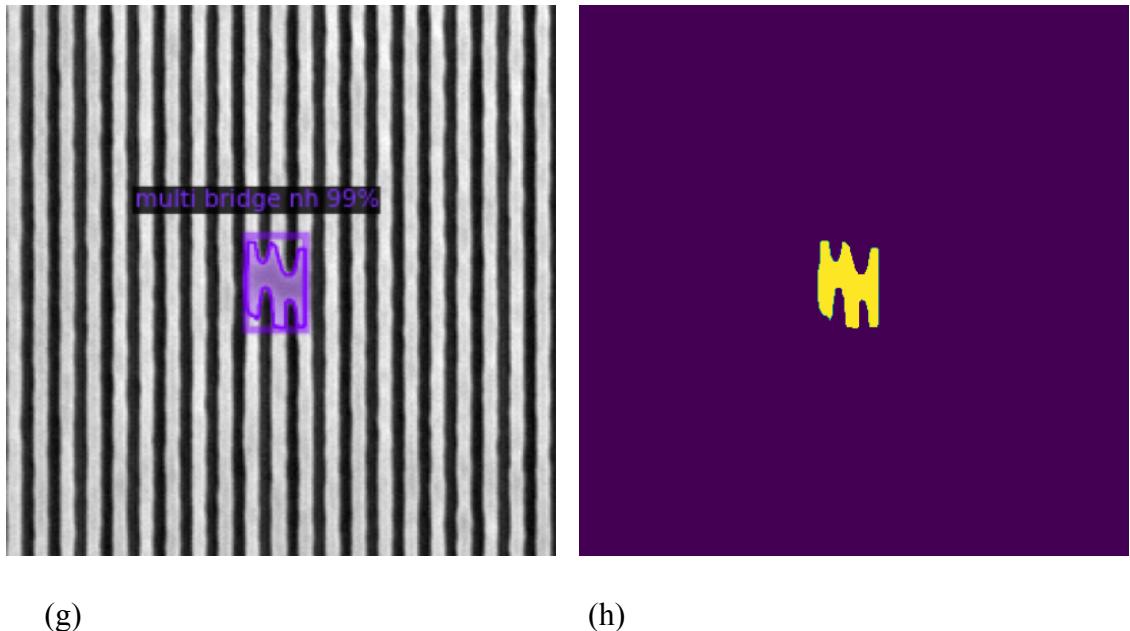
(d)



(e)



(f)



(g)

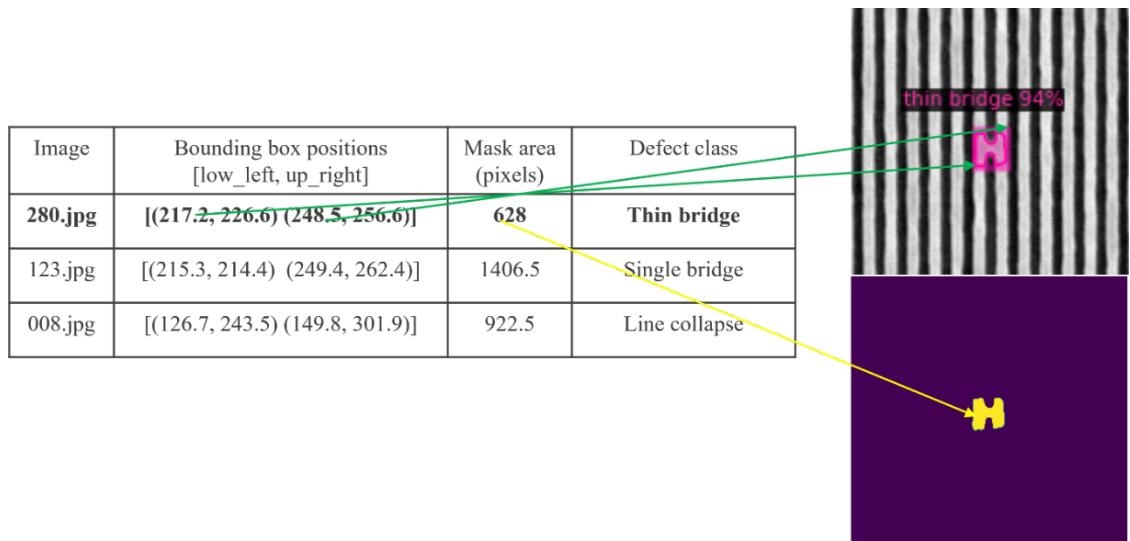
(h)

**Figure 4.17.** Defect detection results with proposed Mask R-CNN approach. (a) Line collapse, (c) Single bridge, (e) Thin bridge, and (g) Multi-line non-horizontal bridge. Output masks for corresponding (b) Line collapse, (d) Single bridge, (f) Thin bridge, and (h) Multi-line non-horizontal bridge defects.

Therefore, proposed method helps to count each categorical defect instances as well as to calculate the surface area in terms of pixels as shown in Fig. 4.18. Table 4.7 provides the comparison analysis for defect detection accuracies obtained per defect class as well as mAP on test images for the Mask R-CNN approach with score-threshold 0.5, with AP50.

**Table 4.7.** Overall test accuracy of proposed Mask R-CNN based ADCD framework. IOU @ 0.50/AP50.

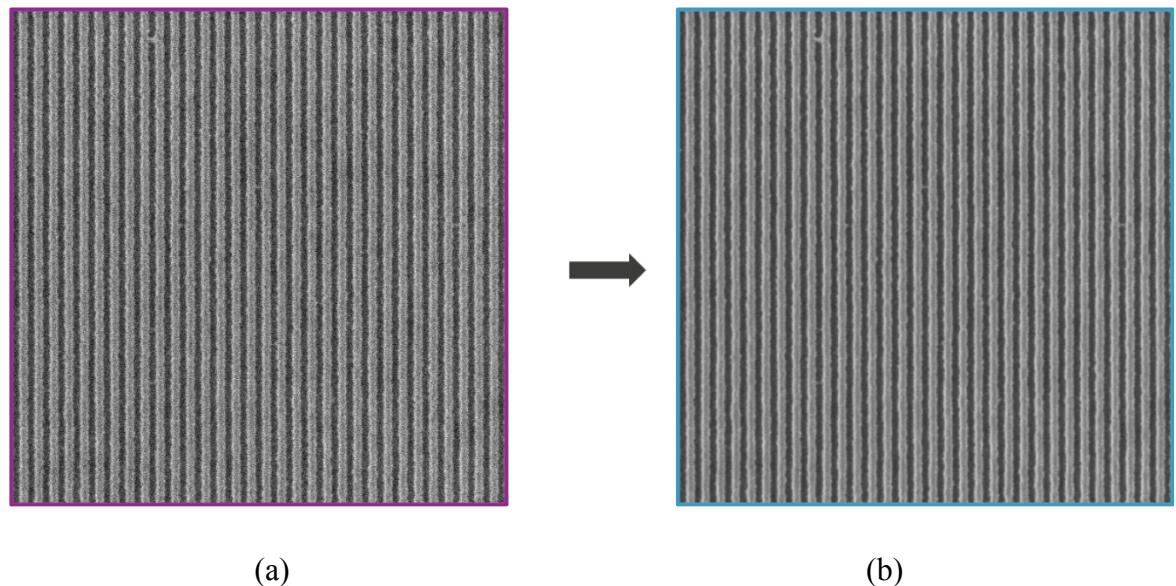
	line_collapse	single_bridge	thin_bridge	multi_bridge_nh	Mean
<b>Bbox AP</b>	0.891	1.00	1.00	0.851	0.936
<b>Segmentation AP</b>	0.891	1.00	1.00	0.851	0.936



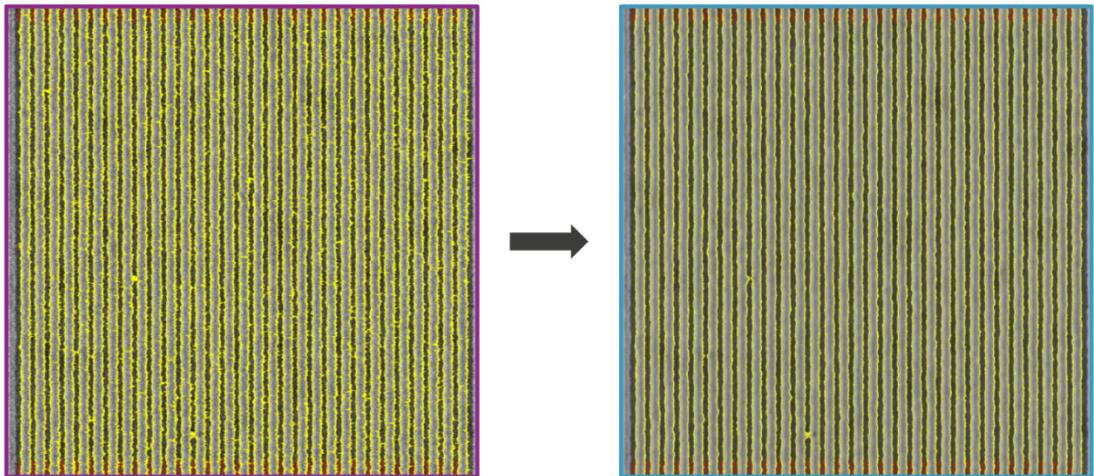
**Figure 4.18.** Defect class prediction with corresponding output mask

#### 4.5 C. Deep Learning Denoiser

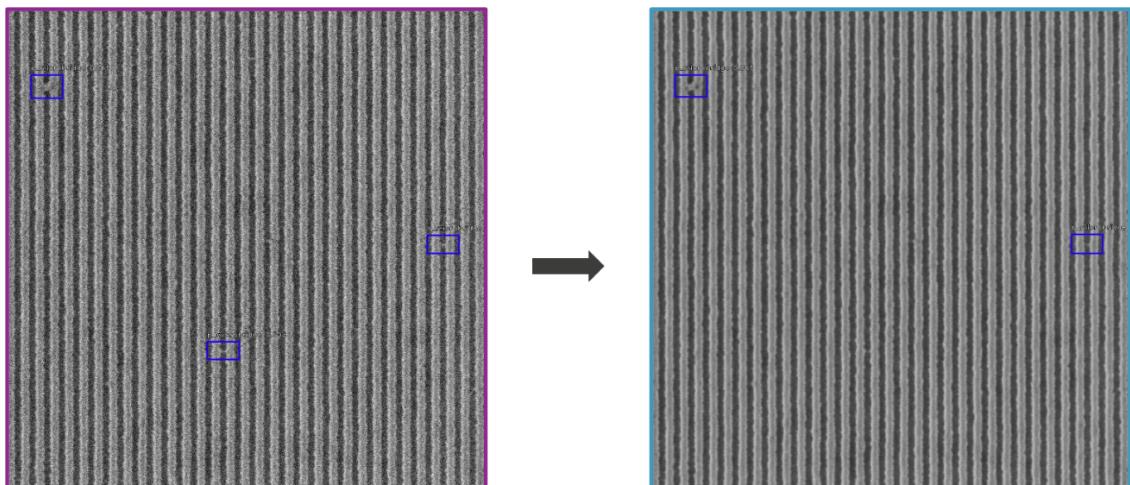
We have demonstrated how denoising can improve defect inspection performance and accuracy in challenging defect-detection scenarios, specifically in case of micro-bridge detection.



**Figure 4.19.** (a) Noisy SEM image [P32] with micro/nano-bridges (b) Denoised image.



**Figure 4.20.** SEM image analysis with Fractilia MetroLER library (a) Noisy image (b) Denoised image. Detected edges in denoised image are with less spikes or almost without spikes in comparison to noisy image.



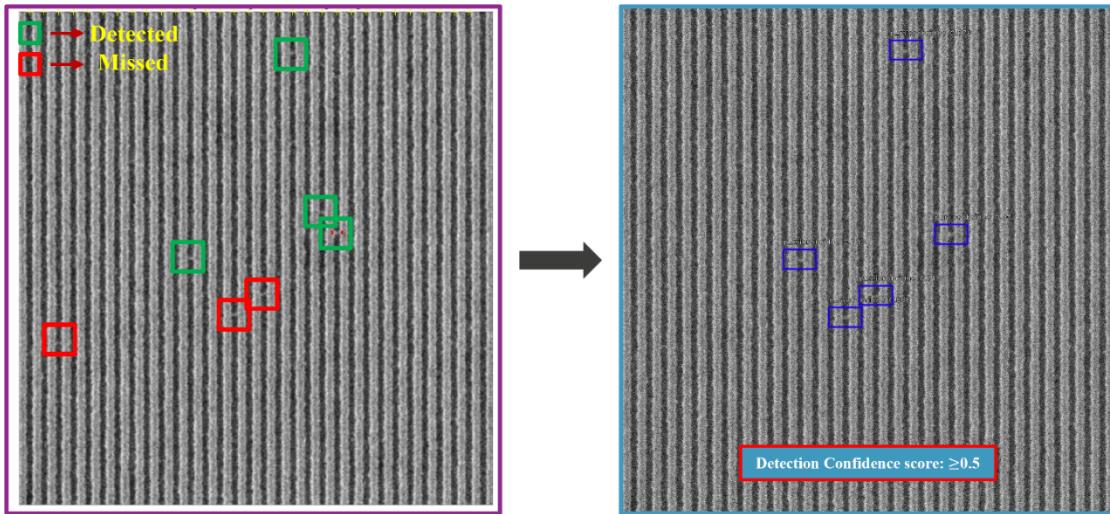
**Figure 4.21.** Defect inspection on (a) Noisy SEM image [P32] with micro/nano-bridges, (b) Denoised image. Denoising generally helps to remove FP defects (resist footing may appear as micro-bridge defects in presence of stochastic noise on structure pixels).

The extraction of repeatable and accurate defect locations along with CD metrology becomes significantly complicated in ADI SEM images due to continuous shrinkage of circuit patterns (pitches less than 32 nm). The noise level of SEM images may lead to false defect detections and erroneous metrology. Hence, reducing noise in SEM images is of utmost importance. In Fig. 4.19, we have shown the denoised SEM image (pitch 32 nm) obtained from the proposed denoising approach. Fig. 4.20 shows detected edges for

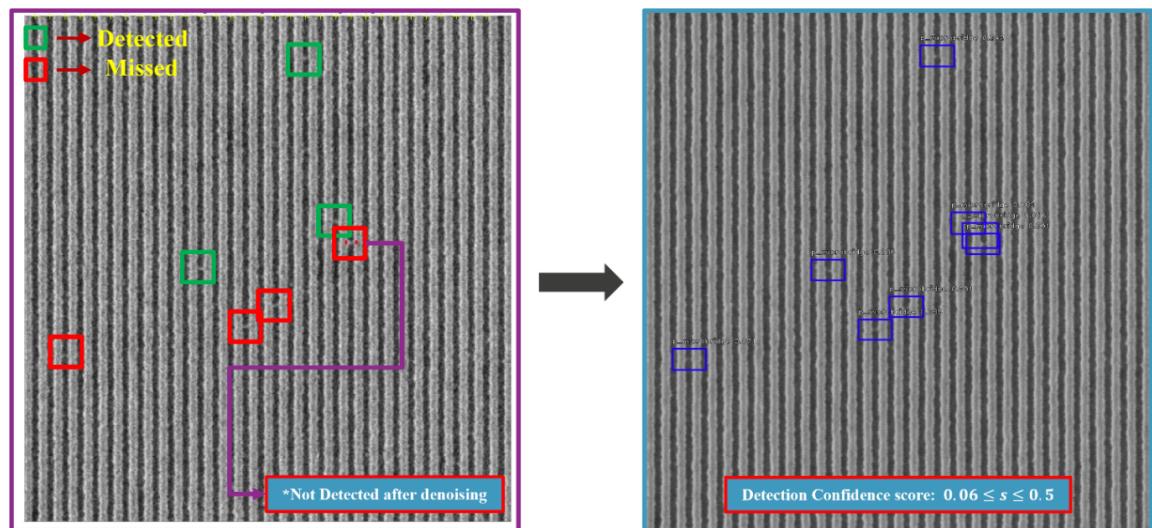
denoised image are with less spikes or almost without spikes in comparison to the noisy SEM image when analyzed with Fractilia MetroLER v2.2.5.0 [53]. While inspecting a noisy SEM image for microbridge detection, both conventional approach [16] and our proposed deep learning based approach may flag false positive (FP) defects in terms of resist footing. In the presence of stochastic noise on structured pixels, resist footing generally appears as tiny microbridges that are expected to be removed during next etch process step. Denoising optimizes this effect of stochastic noise on structured pixels and therefore, helps to remove the false-positive defects (FP) for better metrology and enhanced defect inspection as demonstrated in Fig. 4.21. We have shown two different strategies in this research as (1) remove any FP detection with strict defect detection confidence score  $\geq 0.5$  for microbridge and (2) adaptation of resist footing as “weak microbridge” defect by lowering enough the confidence score ( $0.0 \leq score \leq 0.5$ ). In Fig. 4.22 and Fig. 4.23, we have presented both approaches. For the first approach, we have repeated the defect inspection step on denoised images with the same trained model parameters with noisy images only, whereas for the later, we have retrained the model with denoised images and fine-tuned the model parameters. Another approach is possible as labeling of resist footing as a new defect category and training the model. This will be considered as our future scope of this research.

#### **4.5 D. Comparison with Conventional Approach/Tools**

Moreover, we have presented a comparative analysis on stochastic defect detection performance between our proposed deep learning based approach and conventional approach [16]. Fig. 4.22 provides challenging micro/ nano-bridges detection scenario on the same Noisy L/S SEM image.



**Figure 4.22.** Defect detection on same Noisy SEM image [P32] with micro/nano-bridges: (a) Conventional Tool/approach, (b) Proposed ensemble model based approach. Reprinted with permission from [84].



**Figure 4.23.** Defect detection on same Denoised SEM image [P32] with micro/nano-bridges: (a) Conventional Tool/approach, (b) Proposed ensemble model based approach. Reprinted with permission from [84].

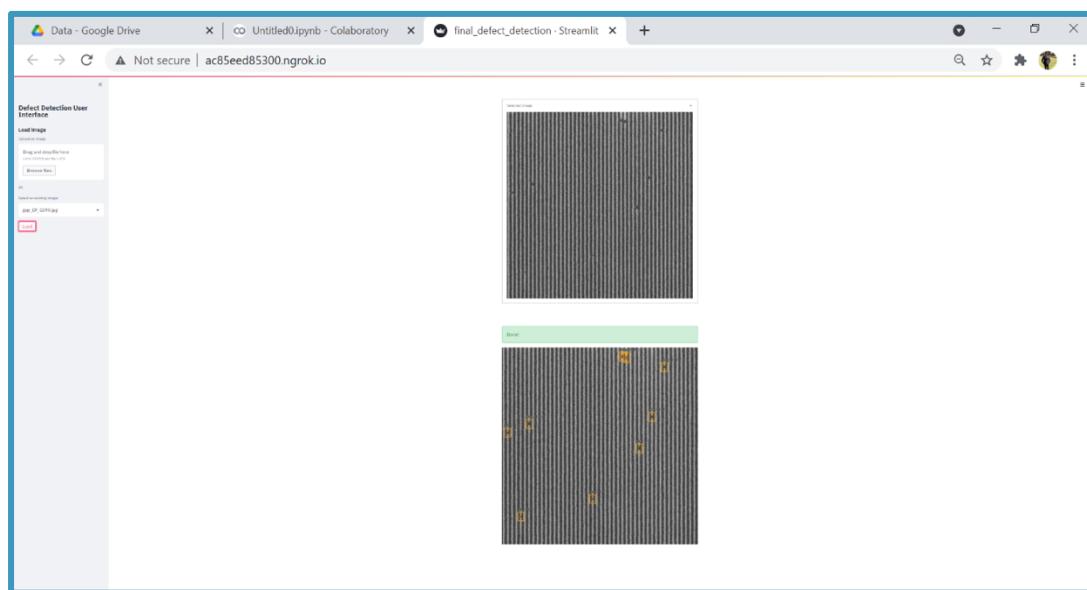
With a “manual” selection of the detection threshold parameters (such as user-defined intensity-threshold, failure size parameter, noise etc.), the conventional approach was able to flag four out of seven observable defects. Whereas, our proposed deep learning based

model automatically detects five out of the same with a strict defect detection confidence score  $\geq 0.5$  without any requirement of such manual trial-and-error based “threshold” selection method. Lowering the automated “confidence score” certainly flags other missing defects as demonstrated in Fig. 4.23. Fig. 4.23 demonstrates the same challenging micro/nano-bridges detection scenario on the corresponding denoised image. We can see the detection scenario is influenced by the condition if the image is noisy or denoised for conventional approach. Furthermore, after denoising, along with previous undetected defect instances, the conventional approach was not able to detect the “most obvious” microbridge defect instance which was flagged before. However, our proposed model demonstrates “stable” performance in detecting defects with better accuracy for both noisy or denoised images and replaces the manual trial-and-error based “threshold” selection method with automated “confidence score”. Once defects are correctly detected, different parameters (as length, width, area, additional feature vectors) about the defects can be output for better understanding the root cause of the defects, as shown in Figs. 4.17, 4.18, respectively. Thus, our proposed approach demonstrates its effectiveness both quantitatively and qualitatively.

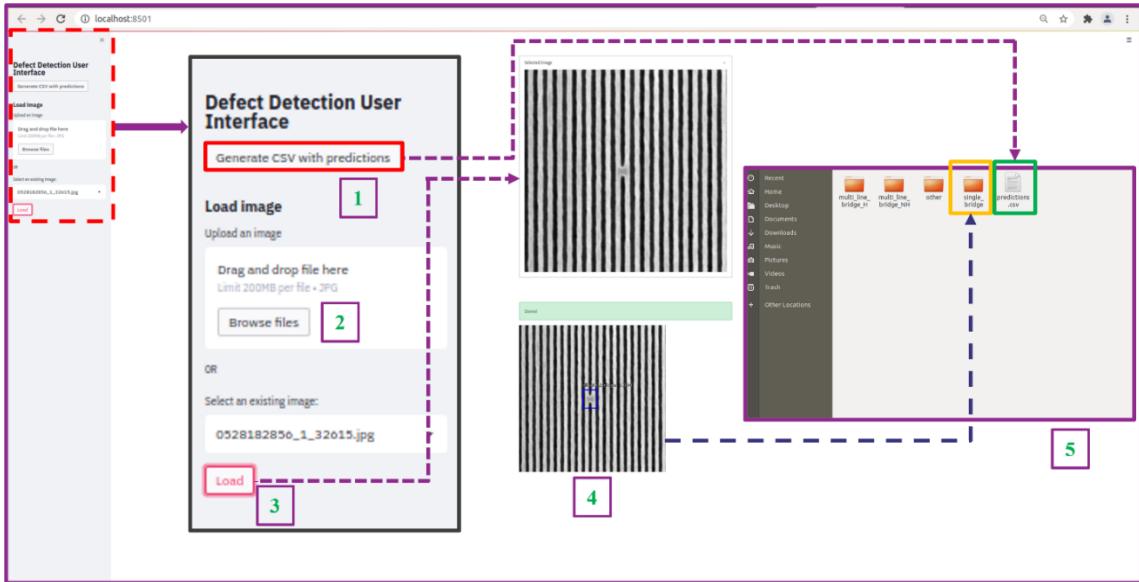
#### 4.5 E. Defect Classifier User-Interface (UI)

We built an UI (User-Interface) using Streamlit library in python script to deploy our proposed model as a web-based defect inspection app. A view of the application interface is depicted in Fig. 4.24. This is a template version of the original proposed software interface and we will add more user-friendly graphical widgets in near future. This will enable different partners/vendors to run the application on their local servers/workstations on their own tool data. This UI will enable the users to upload a dataset of SEM/EDR/Review-SEM images, to select and run one out of different defect detection

inference models on the dataset, to visualize the prediction performance locally and finally to segregate and save the images in different folders according to their defect categorical classes in local machines. As shown in Fig. 4.25, the essential components of our proposed web-based defect inspection app functionality are as: (1) this graphical widget (“*Generate\_CSV\_with\_predictions*”) enables the users to automatically analyse multiple wafer/DOE data with a single click. Once the defect inspection analysis will be finished, a csv file will be generated containing different parameters (location, area, length, width, corresponding defect class) of the defects for the corresponding wafers and finally to segregate and save the images in different folders according to their defect categorical classes in local machines as shown in (5). In addition, the graphical widget (2) (“*Browse\_files*”) allows users to manually browse/load specific wafer data folder of interest, whereas graphical widget (3) (“*Load*”) allows users to manually load and inspect individual image files as shown in (4). Therefore, our proposed web-based defect inspection app helps reduce engineering time as well as tool cycle time against manual inspection method associated with defect inspection process.



**Figure 4.24.** Web-based defect inspection app.



**Figure 4.25.** Essential components of web-based defect inspection app functionality. Reprinted with permission from [84].

## 4.6. Summary

In this chapter, we have developed a novel robust supervised deep learning training scheme to accurately classify as well as localize different defect types in SEM images with high degree of accuracy. Our proposed ensemble model is based on a permutation of ResNet50, ResNet101 and ResNet152 architectures as backbones and an experimental selection of preference-based ensemble strategy to combine the output predictions from different models and achieve better performance on classification and detection of defects. Our model demonstrates not only classification of different defect categories as bridges, line-collapses, gaps, micro-bridges, and micro-gaps as well as variable degrees of pixel level defect scenarios for each of these categories. We also have accurately regressed the region-of-defects with a bounding box (represented by center coordinates of box, width, and height) with a detection confidence score. We have also demonstrated the application of

Mask-RCNN for improved defect instance segmentation in SEM images with precise extent of defect as well as generating a mask for each defect category/instance. Furthermore, we have applied an unsupervised machine learning strategy to denoise the SEM images without the requirement of any clean ground truth or synthetic clean SEM images to remove the False-Positive defects and optimize the effect of stochastic noise on structured pixels. PSD (Power-Spectral-Density) analysis demonstrates that only high frequency component related to noise is affected as expected, keeping the low frequency component, related to the actual morphology of the device feature unaltered. Therefore, our proposed defect inspection pipeline demonstrates enhanced defect detection performance, based on detection accuracy without altering the L/S dimensions in aggressive pitches. As circuit patterns are shrinking to accommodate Moore's law, the conventional defect inspection procedures are becoming less effective and often leads to false defect detections and erroneous metrology. Our deep learning-based model demonstrates it can overcome the limitations while improving classification, detection, and localization of different defect categories with higher accuracy. Our future strategy is to extend this work towards (1) generate defect classes and locations, (2) generate parameters for each and every defect, (3) use data to model defect transfer from litho to etch, and finally (4) expand to other SEM applications (Logic/CH structures) as well as use other sets of images as TEM/AFM etc. Another possibility is (1) experimentation with (a) different other state-of-the-art deep feature extractor networks as backbones as well as (b) different recent detector frameworks as YOLOv5, EfficientNet, EfficientDet etc. to further improve the overall mAP accuracy as well as individual defect class mAP accuracy, (2) addition of new defect categories and experiment with fine tuning of the network parameters to further improve the overall mAP metric.

## **Chapter 5: Unsupervised Machine Learning based SEM Image Denoising for Robust Contour Detection\***

Contour detection of an object is a fundamental computer vision problem in image processing domain. The goal is to find a concrete boundary for pixel ownership between an OOI (object-of-interest) and its corresponding background. However, contour extraction from low SNR SEM images is a very challenging problem as different sources of noise shadow the estimation of underlying structural geometries. As device scaling continues to 3nm node and below, the extraction of accurate CD contour geometries from SEM images especially ADI (after developed inspection) is of utmost importance for a qualitative lithographic process as well as to verify device characterization in aggressive pitches. In this chapter, we have applied a U-Net architecture based unsupervised machine learning approach for de-noising CD-SEM images. Unlike other discriminative deep-learning based de-noising approaches, the proposed method does not require any ground-truth as clean/noiseless images or synthetic noiseless images for training. Simultaneously, we have also attempted to demonstrate how de-noising is helping to improve the contour detection accuracy. We have analyzed and validated our result by using a programmable tool (SEMSuite<sup>TM</sup>) for contour extraction. We have de-noised SEM images with categorically different geometrical patterns such as L/S (line-space), C/H (contact-hole), pillars with different scan types etc. and extracted the contours in both noisy and de-noised images. The comparative analysis demonstrates that de-noised images have higher confidence contour metric than their noisy twins, while keeping the same parameter settings for both data input. When the ML algorithm is applied, the contour extraction results would have higher confidence numbers comparing with the ones only applied the conventional

---

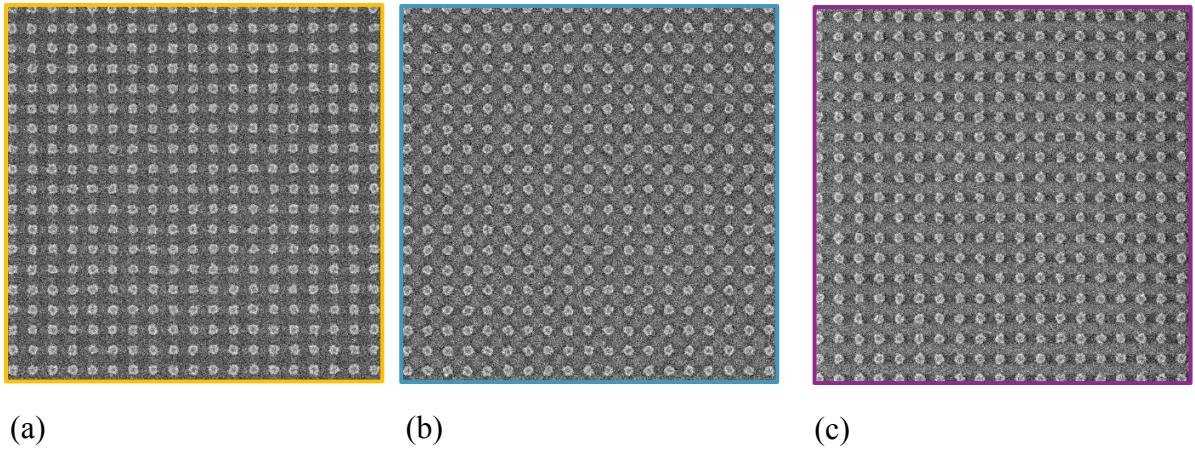
\*Part of the data reported in this chapter is reprinted with permission from Bappaditya Dey, Stewart Wu, Sayantan Das, Kasem Khalil, Sandip Halder, Philippe Leray, Samir Bhamidipati, Kiarash Ahi, Mark Pereira, Germain Fenger, Magdy A. Bayoumi, "Unsupervised machine learning based SEM image denoising for robust contourdetection," Proc. SPIE 11854, International Conference on Extreme Ultraviolet Lithography 2021, 1185411 (12 October 2021); <https://doi.org/10.1117/12.2600945>.

Gaussian or Median blur de-noise method. The final goal of this chapter is to establish a robust de-noising method to reduce the dependency of SEM image acquisition settings and provide more accurate metrology data for OPC calibration.

### **5.1. Introduction:**

As EUV based lithography gets adopted to keep scaling semiconductor devices in our chips, new metrology and inspection challenges arise. We need to measure these small dimensions fast but without losing accuracy and repeatability. Metrology and inspection are at the heart of process control. Without adequate metrology and inspection capability yields suffer. So, as we continue to shrink the pitch (below 36nm), measurement of dimensions and classification of defects become a key issue. There have been numerous attempts to standardize measurement LER/LWR methods [85]. Roughness and low thickness of EUV resists, pose significant challenges especially after development (ADI). SEM images are usually noisy and low contrast as (*a*) low  $k_1$  factor limited lithography process, (*b*) a smaller number of frames are used to avoid resist shrinkage. Fig. 5.1 shows example of three different noisy SEM images for orthogonal pillar array patterns, which contains different non-uniform background intensity produced by different SEM scanning mode intentionally. Conventional contour extraction techniques, such as Sobel operator, gradient operator, Canny edge detector etc., do not survive the requirements of the semiconductor industry at these advanced nodes. Though applying these traditional methods have limited advantages in terms of simple scripting and speedy computation only, in contrast, the major disadvantages are: (*a*) existence of stochastic noise significantly impacts contour extraction itself, (*b*) accuracy of an extracted contour cannot be guaranteed in terms of width of a single pixel and therefore may appear either as (1) edge of an isolated one or, (2) a small continuous section. In this chapter, we have revisited our machine

learning based denoising method [56] for improved contour extraction/detection of SEM images. We have shown that by reducing the noise in SEM images without altering the PSD we can extract contours accurately.



**Figure 5.1.** Noisy CD-SEM images [Orthogonal pillar array (design CD=20nm, pitch=40nm) with 3 different scanning directions]: (a) Scan\_1, (b) Scan\_2, (c) Scan\_3. Reprinted with permission from [76].

We have addressed the following problems: (1) denoising only a single noisy SEM image acquisition for each of categorically different geometrical patterns such as L/S (line-space), C/H (contact-hole), pillars with different scan types, without requirement of any clean ground-truth in true sense. (2) to demonstrate, how the applied deep learning based denoising algorithm is helping to improve the contour detection/extraction capability/accuracy, which in turn, allows better identification of defects. (3) Analysis and validation of our result by using a programmable tool (SEMSuite<sup>TM</sup>) for contour extraction.

The comparative analysis demonstrates that de-noised images have higher confidence contour metric than their noisy twins while keeping the same parameter settings for both data input. Therefore, our unsupervised approach demonstrates its effectiveness both quantitatively and qualitatively.

## **5.2. Related Work:**

We briefly investigated some existing recent research approaches and methodologies in the context of extracting SEM image contours. In [86], the authors presented a method to evaluate and optimize the critical dimension (CD) matching between a reference CD and extracted SEM contours. The authors reported to achieve a 3s-total measurement uncertainty of 0.8 nm and 3.2 nm for 1D and 2D patterns, respectively, by optimizing some matching calibration parameters as well as the selection of the best anchor pattern for the corresponding matching. [87] proposed a robust and efficient framework to enable the automation of SEM image analysis and therefore to analyse the variability of process. The authors proposed (*a*) a new effective noise reduction filter, (*b*) an algorithm to detect the topography inside a SEM image, (*c*) applying phylogenetic analysis for rapid identification of analogous prints categories inside large group of images. A CD-gap-free contour extraction technique has been proposed in [88] for OPC model calibration. The terminology “CD-gap” represents the deviation in calculation between reference CD and CD measurements calculated from SEM contours. To reduce this error, the authors first classified mask-edges into corresponding shape structures and then extracted an optimized SEM contour for those shape structures. [89] presented a machine learning based approach for contour extraction of SEM images. They have designed a Deep Convolution Neural Network (DCNN) for the contour extraction purpose. Before training the neural network, they applied another neural network for denoising the original noisy SEM images.

Furthermore, the authors prepared contour training pair of SEM images by applying conventional Canny edge detection technique with few additional image processing methods.

### **5.3. Applied Method:**

As our main focus of this chapter is to demonstrate how our applied deep learning based denoising algorithm is helping to improve the contour extraction/detection capability/accuracy with CD-SEM images, the network is trained following the identical scheme as described in [56, 51]. The applied denoising approach is based on Noise2Void training scheme introduced by Krull *et al.* [51]. The goal of the applied deep learning (DL) denoiser is to recover the clean signal part ( $s$ ) from a noisy image ( $x$ ), by means of only removing/separating the signal degrading noise part ( $n$ ), where:

$$x = s + n \quad \text{Eq. 5.1}$$

The applied method, in the context of any noisy SEM image, is based on assumptions, as:

$$p(s_i|s_j) \neq p(s_i) \quad \text{Eq. 5.2}$$

$\Rightarrow$  neighborhood pixel values for  $s$

$= (s_1, s_2, \dots, s_i, s_j, \dots s_n)$  are statistically dependent

and

$$p(n|s) = \prod_i p(n_i|s_i) \quad \text{Eq. 5.3}$$

$\Rightarrow$  pixel values for  $n$

$= (n_1, n_2, \dots, n_i, n_j, \dots n_n)$  are conditionally independent given corresponding signal

$$values \text{ for } s = (s_1, s_2, \dots, s_i, s_j, \dots s_n)$$

Therefore, any image  $x$  can be realized as a joint probability distribution function

$$p(s, n) = p(s)p(n|s) \quad \text{Eq. 5.4}$$

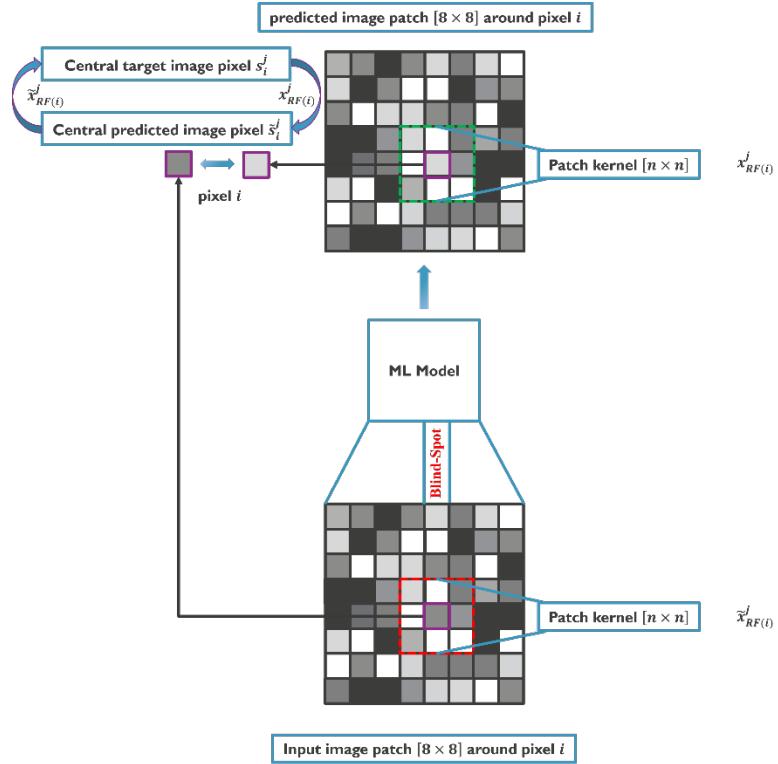
$$\text{Therefore, } E[x_i] = s_i \rightarrow GT[\text{noiseless}_i\text{mage}], \text{ iff } E[n_i] = 0 \quad \text{Eq. 5.5}$$

We have used single noisy SEM image for each categorically different geometrical patterns such as L/S (line-space), C/H (contact-hole), pillars with different scan types (as Scan\_1 type, Scan\_2 type and Scan\_3 type) for training of our algorithm without any such requirement of GT (ground-truth) or synthetic noiseless images. The applied deep learning denoiser network is shown in Fig. 5.2. This framework allows to extract the input patch as well as the target patch from the same noisy training image. Training patches of size  $[64 \times 64]$  have been extracted randomly from each single noisy SEM image of the above-mentioned pattern types. The blind-spot masking mechanism, as demonstrated in Fig. 5.3, has been implemented to derive both the input and target from the same noisy training image, and therefore, leverages unsupervised training approach. This mechanism also enables inability to learn the identity between an input pixel  $\tilde{s}_i$  to its corresponding target pixel  $s_i$ . However, the model is able to estimate clean signal part  $s_i$  by analyzing its neighborhood in close proximity within a certain radius (following Eq. 5.2 as statistical dependency) but carry no information about the noise part (following Eq. 5.3 as statistically pixel-wise independent). For a receptive field  $\tilde{x}_{RF(i)}^j$  of a training patch with blind-spot at the center pixel  $i$ , the goal is to minimize pixel-wise loss between input patch and target patch by fine tuning the weight-parameters  $\varphi$  as:

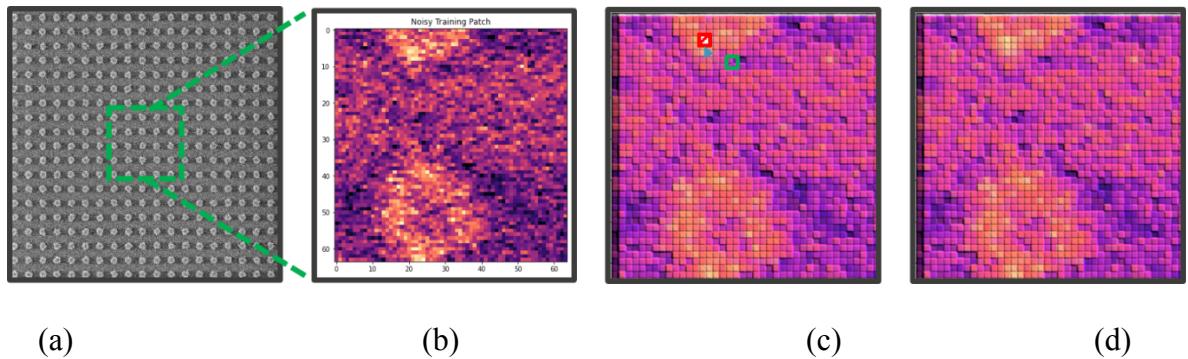
$$\arg \varphi \min \sum_j \sum_i J(g(\tilde{x}_{RF(i)}^j; \varphi), x_{RF(i)}^j) \quad \text{Eq.}$$

5.6

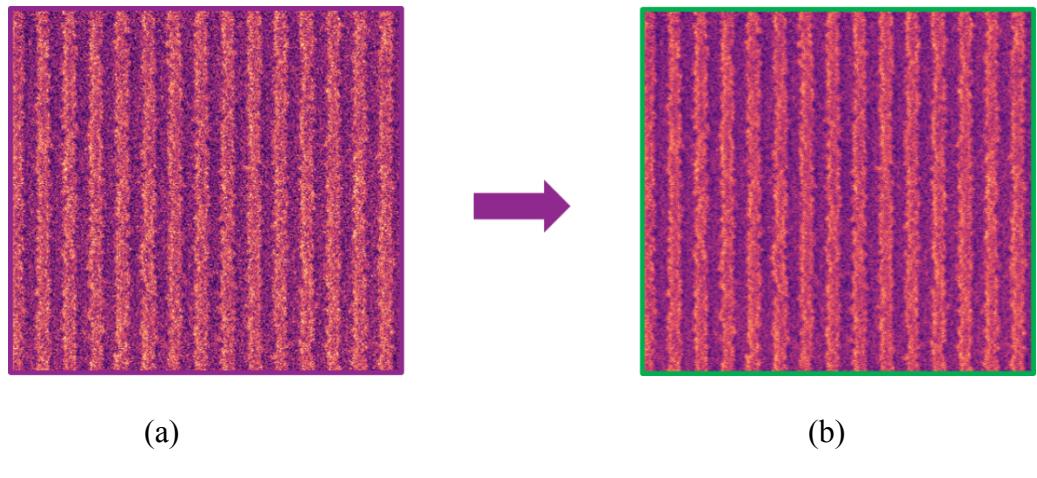
In Fig. 5.4, Fig. 5.5, and Fig. 5.6, we have shown the results obtained from the applied denoising approach for each categorically different geometrical pattern. Fig. 5.4 (c) shows the line-scan plot comparison for noisy/denoised image pair for Pitch 28nm L/S pattern with film thickness 8 nm [Thin Resist] as analyzed with ImageJ library [90]. As two line-scans are overlapping one another (only noise-spikes are clipped in denoised image), this indicates that the actual information content of the image was not degraded by the applied denoising approach. We have analyzed each pair of original noisy image and its corresponding denoised image with Fractilia MetroLER v2.3.0.0. library [53]. In Fig. 5.4 (d), we have extracted and aligned SEM contours GDS [cropped ROI] with Fractilia MetroLER library for noisy/denoised image pair for thin resist L/S pattern, which again follows our ImageJ library analysis. We have also extracted, analyzed and compared different significant metrics (as Power Spectral Density, LCDU, defectivity, LER, LWR, CD, Pitch etc.) for each pair of original noisy image and its corresponding denoised image, appropriate for each categorically different geometrical pattern. This study enables us to demonstrate that only the high frequency component related to noise is clearly affected, as expected, while the low frequency component, related to the actual morphology of the feature, is unaltered so that edge placement errors (EPE) are not changing. This also implies that the information content of the denoised images was not degraded by our applied denoising approach.



**Figure 5.2.** Proposed deep learning denoiser network. Reprinted with permission from [76].

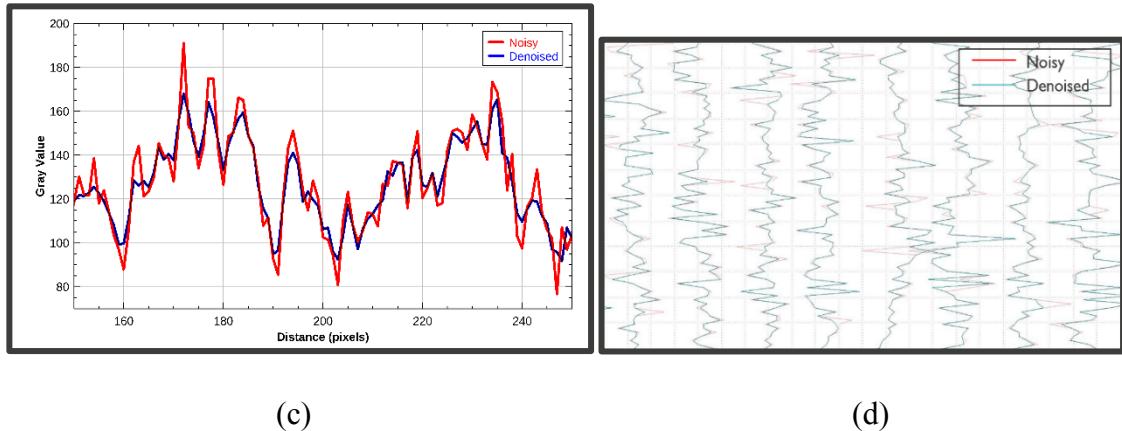


**Figure 5.3.** Blind-spot masking mechanism. (a) Original noisy SEM image. (b) image patch extracted from (a). (c) Training patch: A randomly selected pixel intensity value (labelled green) replacing target pixel intensity value (labelled red) towards achieving blind-spot masking mechanism. This modified image is used as input patch during model training. (d) Target patch: [corresponding to (c)] raw original input patch without any modification. Reprinted with permission from [76].



(a)

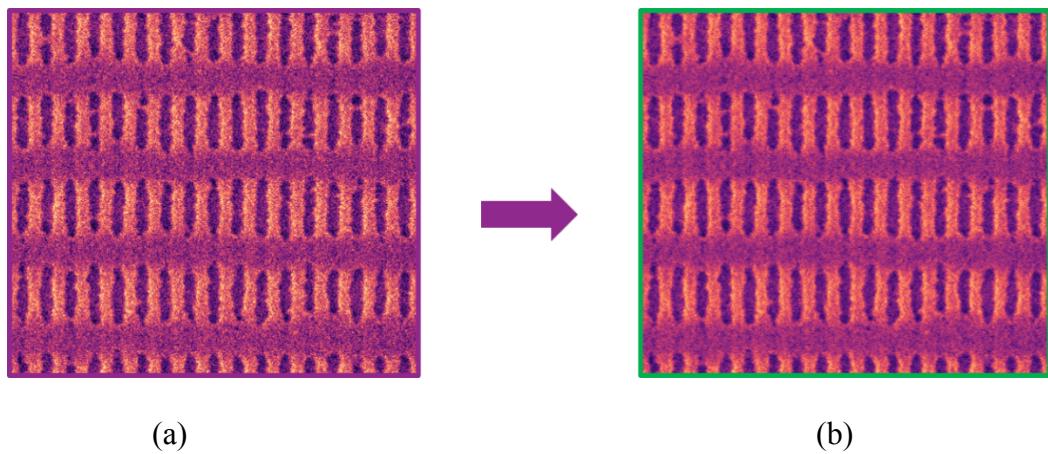
(b)



(c)

(d)

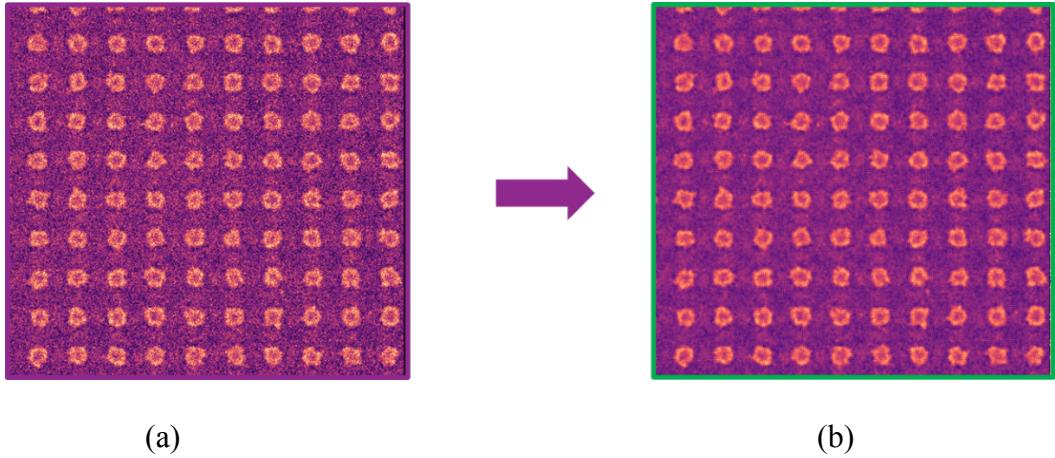
**Figure 5.4.** (a) Noisy SEM image P28 with film thickness 8 nm [Thin Resist] (b) Denoised image (c) Line-Scan plot comparison for both noisy and denoised SEM images ROI. (d) Extracted and aligned SEM contours GDS [cropped ROI] with Fractilia MetroLER library. Reprinted with permission from [76].



(a)

(b)

**Figure 5.5.** (a) Noisy SEM image [T2T CD = 26nm], (b) Denoised image. Reprinted with permission from [76].



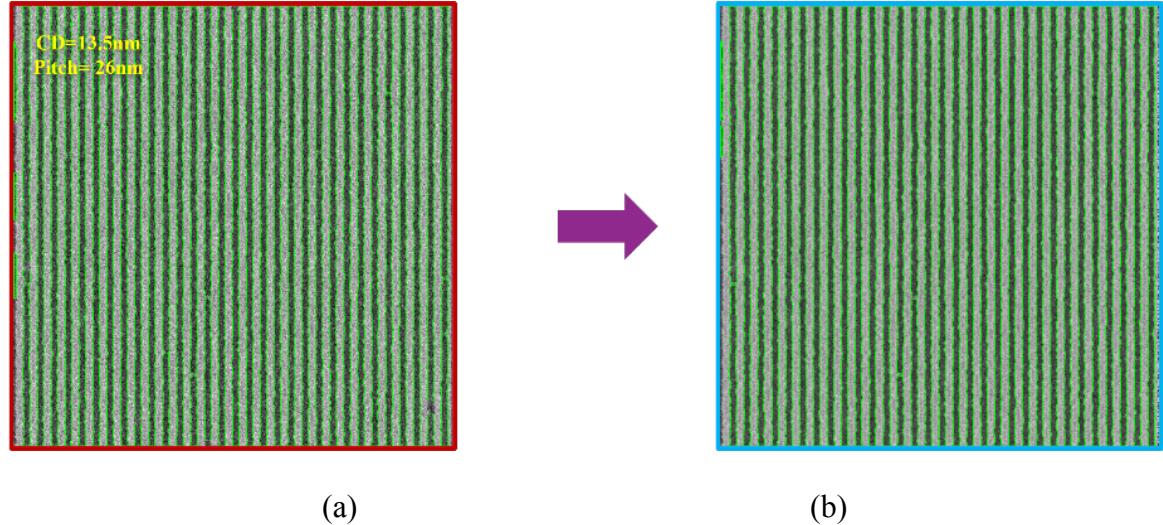
**Figure 5.6.** Orthogonal pillar array [Type: Scan\_1] (design CD=20nm, pitch= 40nm): (a) Noisy image, (b) Denoised image. Reprinted with permission from [76].

#### 5.4. Implementation and Experimental Results

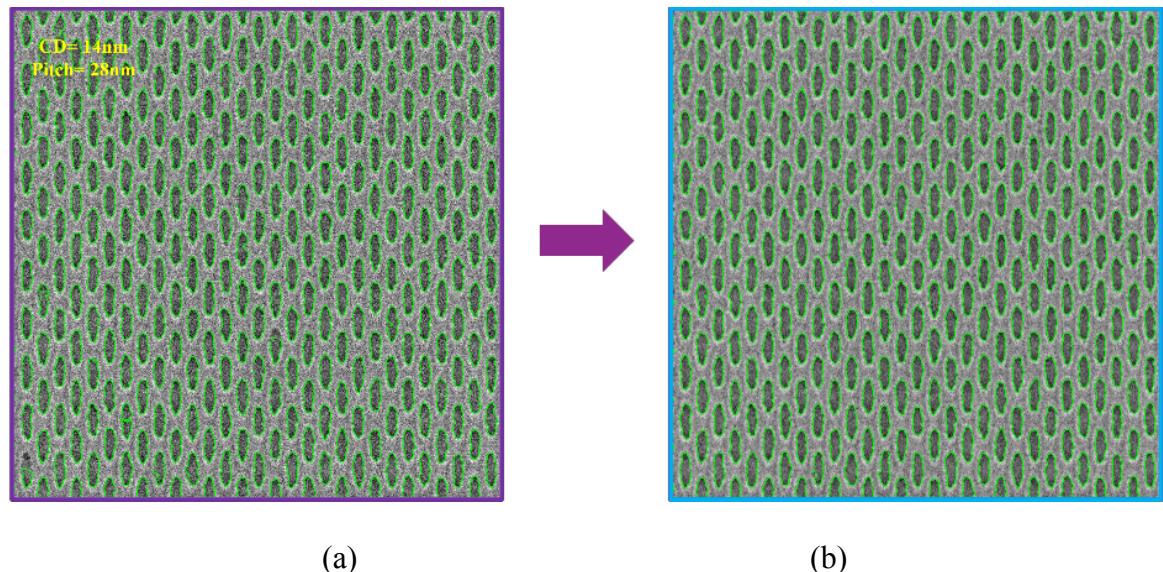
To demonstrate, how the applied deep learning based denoising algorithm is helping to improve the contour detection capability/accuracy, the contours extraction is performed using the same initial image pre-processing setting on Calibre® SEMSuite™ to analyse each pair of original noisy image and the corresponding denoised image. Three different layout types (Line/Space, Contact/Hole, orthogonal Pillar array) are processed and analysed by different indices. In this evaluation of the extracted contours, only the “**high confidence**” contours, labelled in green, is shown without any “low confidence” contour(s). Inside of SEMSuite™, the “high/low confidence” contour is categorized simultaneously. In addition, there are multiple indexes in the metric for further analysis when extracted contour is provided.

##### 5.4 A. Line/Space (L/S) and Contact/Hole (C/H) Layout

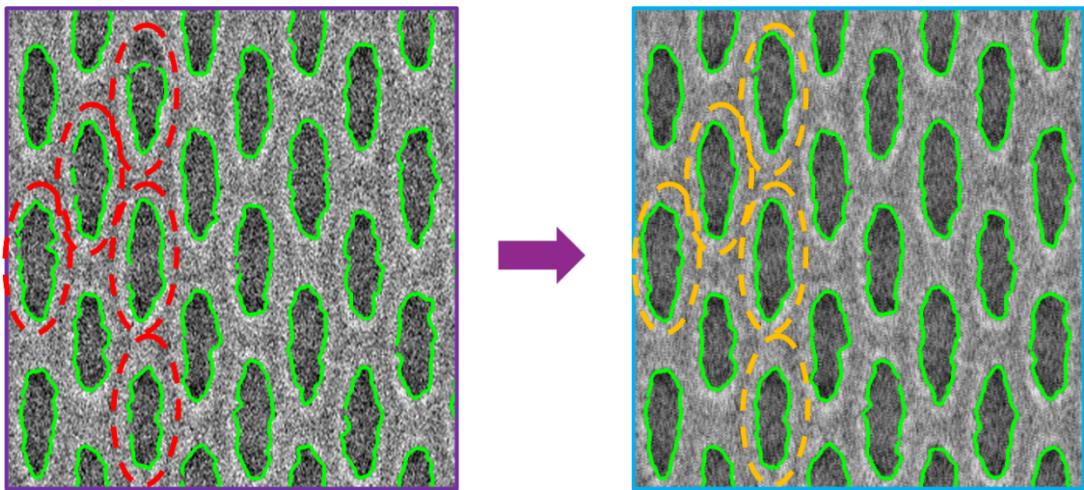
By using an initial image pre-processing setting in SEMSuite<sup>TM</sup>, none of the polygons in the FOV (Field-of-View) were missed, for both noisy and denoised images, across these two pattern types.



**Figure 5.7.** L/S patterns at pitch 26 nm: Contour extraction result analysis on Calibre<sup>®</sup> SEMSuite<sup>TM</sup>. (a) Noisy image (b) Denoised image. Reprinted with permission from [76].

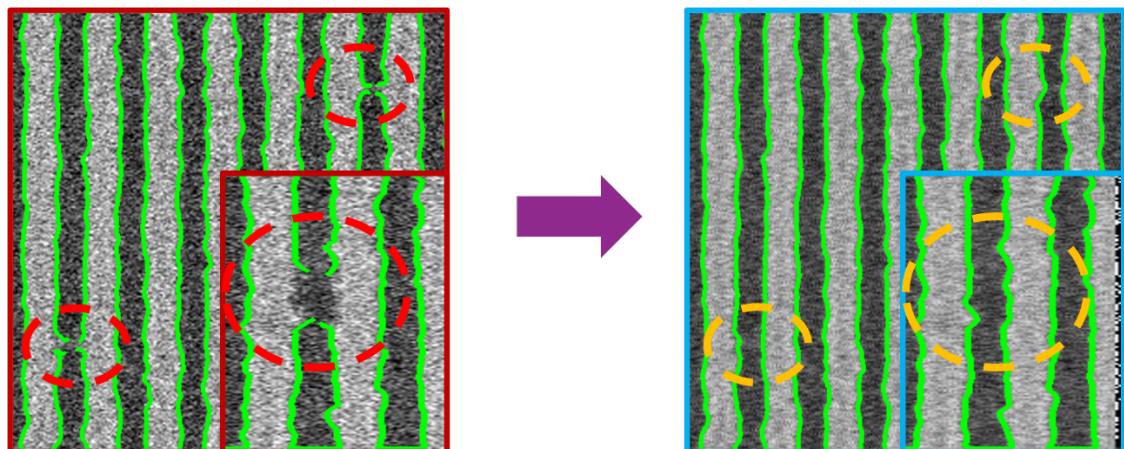


**Figure 5.8.** C/H patterns at pitch 28 nm: Contour extraction result analysis on Calibre<sup>®</sup> SEMSuite<sup>TM</sup>. (a) Noisy image (b) Denoised image. Reprinted with permission from [76].



**Figure 5.9.** Number of partially extracted contours reduced significantly in denoise images than the ones in noise images for Contact/Hole pattern type. Reprinted with permission from [76].

\*\*\* Contours for all patterns are extracted  
 \*\*\* Only 'High Confidence' contours are shown



**Figure 5.10.** Detection of “# of false micro-bridges” reduces significantly in denoise images than the ones in noise images for Line/Space pattern type. Reprinted with permission from [76].

\*\*\* Contours for all patterns are extracted  
 \*\*\* Only 'High Confidence' contours are shown

For C/H scenario, “High Confidence Overall %” is indicative of the ratio of number of green contour points on extracted contour to total number of points on extracted contour,

across all patterns in the image. In Table 5.1, we have shown the impact when the unsupervised algorithm is applied. The “High Confidence Overall %” is relatively higher in denoised image than the raw noisy one even when all the polygons are extracted for both. Fig. 5.7 and Fig. 5.8 shows the contour extraction result comparison for L/S patterns at pitch 26 nm and C/H patterns at pitch 28 nm, respectively. To be specific, “High Confidence Overall %” is indicative of the ratio of number of green contour points on extracted contour to total number of points on extracted contour, across any/all patterns in the images. In Fig. 5.9, we have demonstrated that number of partially extracted contours reduces significantly in denoised image than its noisy twin for contact/hole scenario. For L/S patterns, “# of bridges” reports the micro-bridges that are identified by contour extraction functionality. In addition, these bridges have been categorized as “true” or “false” depending on whether the bridge was caught by contour extraction due to noise or whether there might be indeed an underlying process-based stochastic bridge. In Table 5.2, Fig. 5.10, we have demonstrated that the denoised images help to identify and reduce the “# of false bridges” against the noisy twins, where “# of bridges” are flagged due to the presence of stochastic noise on the body of SEM images.

**Table 5.1.** Summary of analysis metric from SEMSuite<sup>TM</sup> (**Contact/Hole** patterns). Reprinted with permission from [76].

Images (Noisy/Denoised)	Gauge ID	High Confidence Overall %
Noisy Raw Image	1	96.61
Denoised_normal_withoutCP	2	<b>98.85</b>

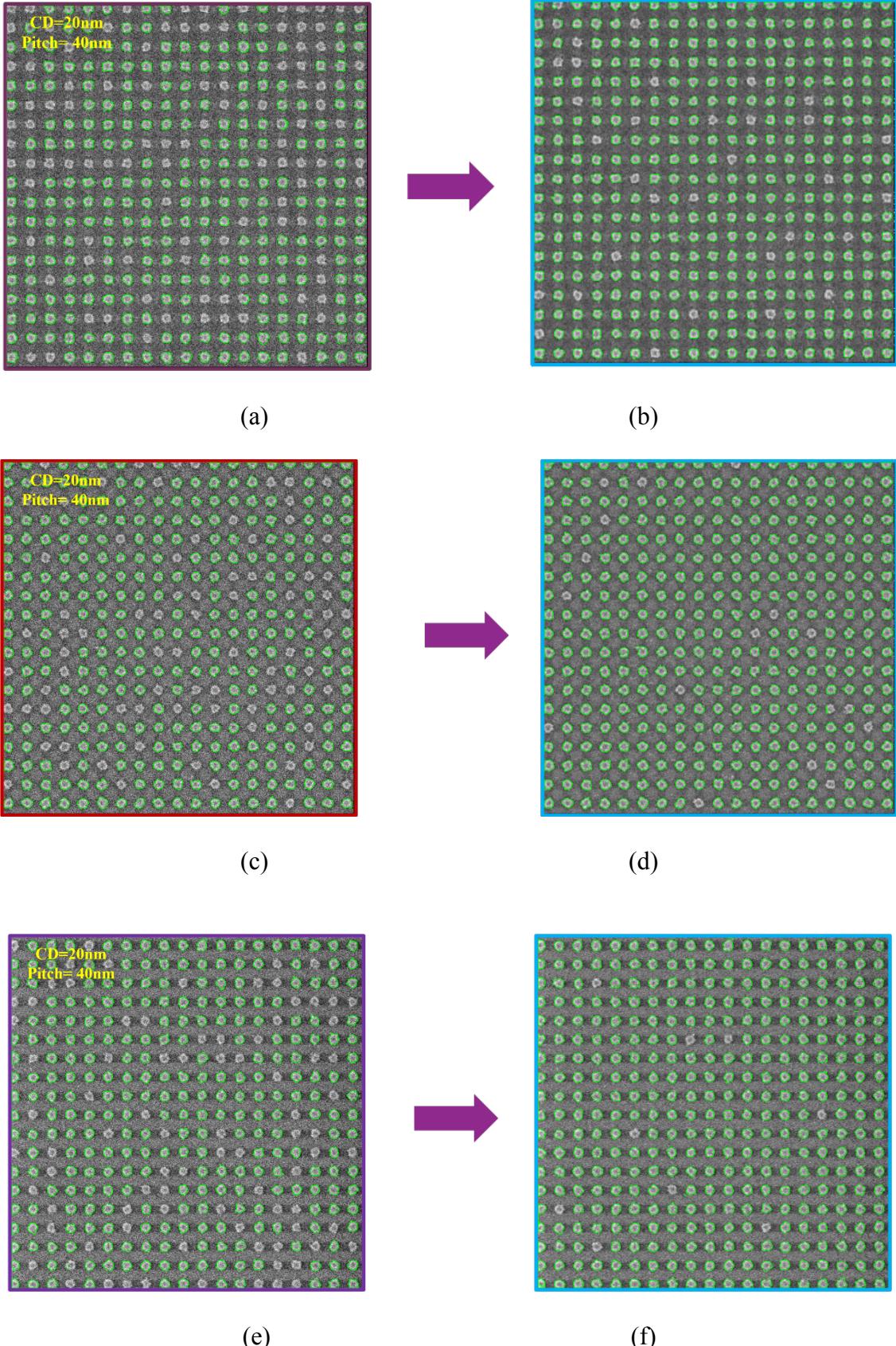
**Table 5.2.** Summary of analysis metric from SEMSuite™ (**Line/Space** patterns).  
Reprinted with permission from [76].

Images (Noisy/Denoised)	Gauge ID	# of Bridges	
		TRUE	FALSE
Noisy Raw Image	1	3	17
Denoised_normal_withoutCP	2	3	0

#### 5.4 B. Pillar Layout With Different Scan Types

For pillar patterns, there are 3 different images/dataset which contains different non-uniform background intensity produced by different SEM scanning mode intentionally.

Fig. 5.11 shows the contour extraction results comparison for Scan\_1 [Fig. 5.11 (a)(b)], Scan\_2 [Fig. 5.11 (c)(d)] and Scan\_3 [Fig. 5.11 (e)(f)] pillar array patterns, respectively, for noisy/denoised image pairs. During training of the model, we created the blind-spot pixels by manipulating neighborhood pixels of an image patch in 4 different experimental ways as (1)  $p1 \rightarrow$  samples neighborhood pixels according to a continuous uniform distribution with the center pixel of the patch kernel, (2)  $p2 \rightarrow$  samples neighborhood pixels according to a normal gaussian distribution without the center pixel of the patch kernel, (3)  $p3 \rightarrow$  sampling a random number from a gaussian distribution with  $mean(\mu) = 0$  and  $standard deviation(\sigma) = radius$  of patch kernel and adding this random number to the original pixel value and (4)  $p4 \rightarrow$  sampling a random number from a gaussian distribution with  $mean(\mu) = mean(neighborhood\_pixels)$  and  $standard deviation(\sigma) = standard deviation(patch\_kernel)$ .



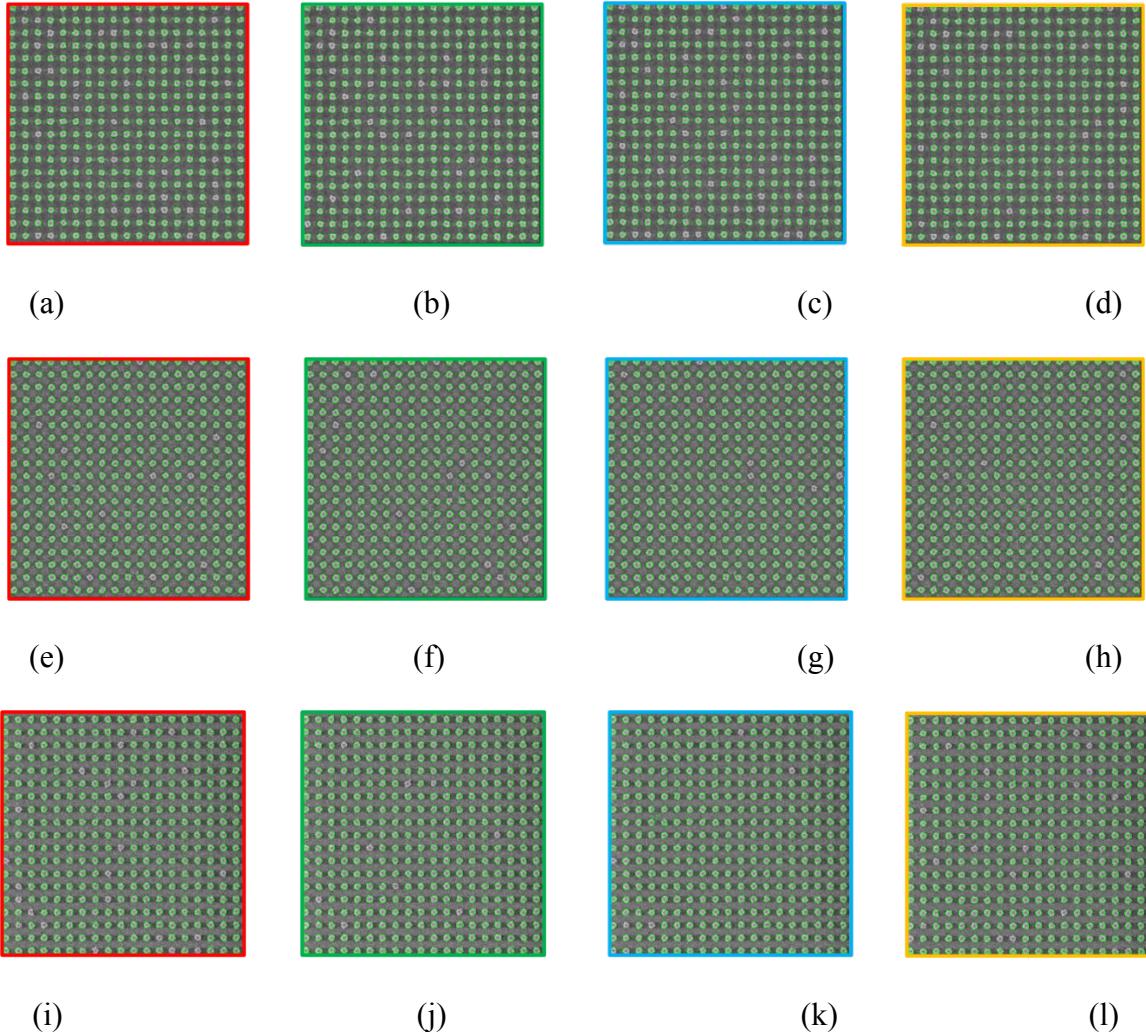
**Figure 5.11.** Orthogonal pillar array (design CD=20nm, pitch= 40nm) with 3 different scanning directions: Contour extraction result analysis on Calibre® SEMSuite™.

Scan\_1: (a) Noisy image (b) Denoised image

Scan\_2: (c) Noisy image (d) Denoised image

Scan\_3: (e) Noisy image (f) Denoised image.

Reprinted with permission from [76].



**Figure 5.12.** Denoised Orthogonal pillar array images with 3 different scanning directions: Contour extraction results comparison on Calibre® SEMSuite™ for four different experimental blind-spot pixel manipulation mechanisms as:

(1)  $p1 \rightarrow \text{uniform\_with\_CP}$

(2)  $p2 \rightarrow \text{normal\_without\_CP}$

(3)  $p3 \rightarrow \text{normal\_additive}$

(4)  $p4 \rightarrow \text{normal\_fitted}$

Scan\_1: (a)  $p1$ , (b)  $p2$ , (c)  $p3$ , (d)  $p4$  [Top Row]

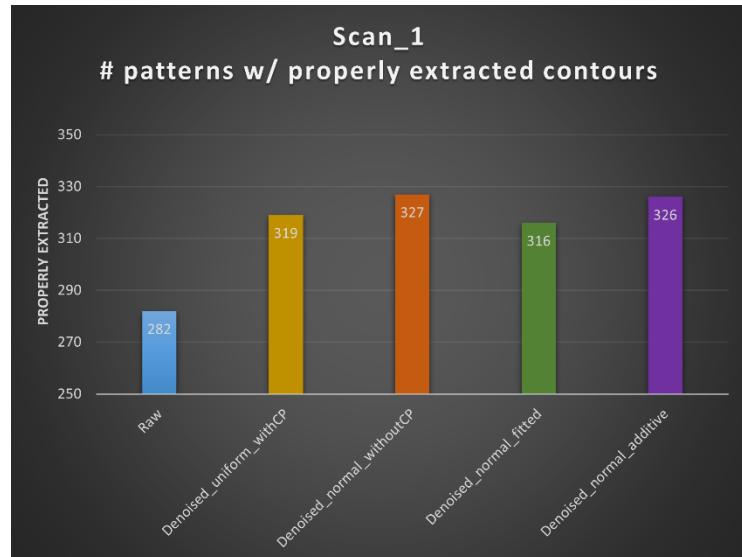
Scan\_2: (e)  $p1$ , (f)  $p2$ , (g)  $p3$ , (h)  $p4$  [Middle Row]

Scan\_3: (i)  $p1$ , (j)  $p2$ , (k)  $p3$ , (l)  $p4$  [Bottom Row].

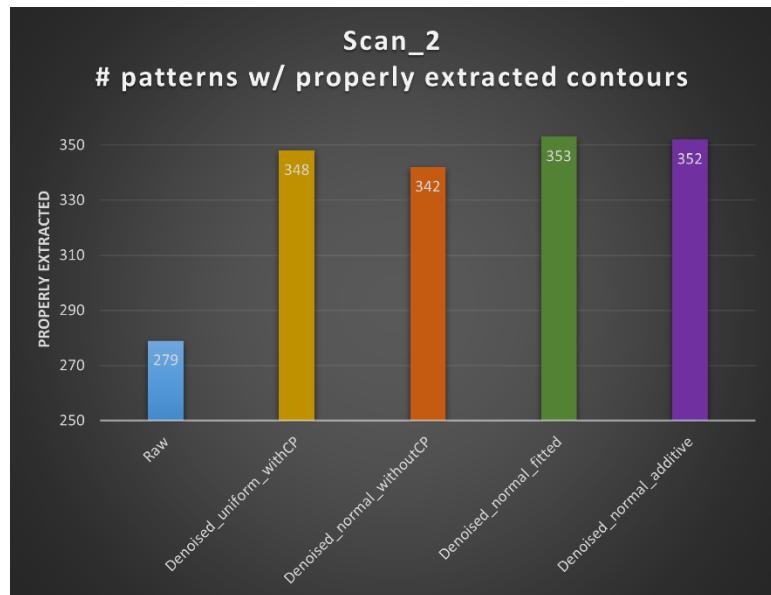
Reprinted with permission from [76].

This is to keep the network away from learning the identity  $f(\mathbf{inp}_{pix} \rightarrow \mathbf{out}_{pix})$  as well as to observe the denoising effect on the contour extraction performance. Fig. 5.12 shows the contour extraction results comparison for four different experimental blind-spot pixel manipulation mechanisms for denoised images as Fig. 5.12 (a)  $p1$ , (b)  $p2$ , (c)  $p3$ , (d)  $p4$  for Scan\_1, Fig. 5.12 (e)  $p1$ , (f)  $p2$ , (g)  $p3$ , (h)  $p4$  for Scan\_2 and Fig. 5.12 (i)  $p1$ , (j)  $p2$ , (k)  $p3$ , (l)  $p4$  for Scan\_3, respectively. Fig. 5.13 depicts the plot of “# patterns with properly extracted contours” metric against the raw noisy and de-noised images with three different scan types as (a) Scan\_1, (b) Scan\_2, (c) Scan\_3 on the Pillar patterns. There are two observables: (a) the number of patterns with fully extracted contours in denoised images are significantly higher than the ones in noise images across all the scanning modes for all four different experimental blind-spot pixel manipulation mechanisms [ $p1, p2, p3, p4$ ] and (b) for each scanning mode, the number of patterns with fully extracted contours in denoised images for all four different experimental blind-spot pixel manipulation mechanisms [ $p1, p2, p3, p4$ ] are in close proximity. Table 5.3, “# missed patterns” indicates the number of patterns in design for which no contour was extracted. This number also significantly reduced in denoised images, as shown in Fig. 5.14 [only shown on  $p2$  across all the scanning modes]. In addition, number of properly/partially extracted contours differs not only between noisy/denoised image pairs, but also across different scanning modes due to varying pixel intensities distribution in the background, as shown in Fig. 5.15 [only shown on  $p2$  across all the scanning modes]. Also, another trend in both Fig. 5.14 and Fig. 5.15 is observed that besides the decreasing/increasing number from noisy image to denoised ones, the slowly decreasing/increasing difference from “Scan\_1” image to “Scan\_3” image is also observed. Therefore, with applied deep learning

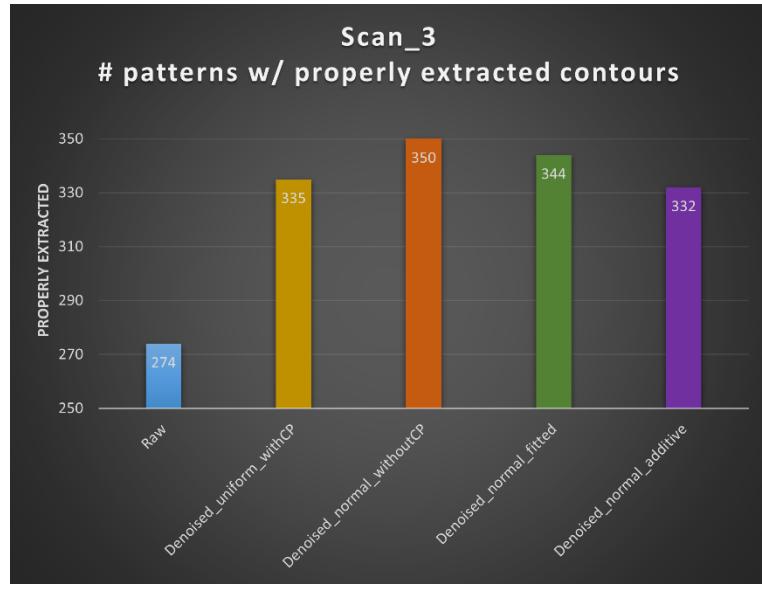
based denoising algorithm, the stochastic noise has been removed which in turn optimizing the varying non-uniform background intensity for better contour detection.



(a)



(b)



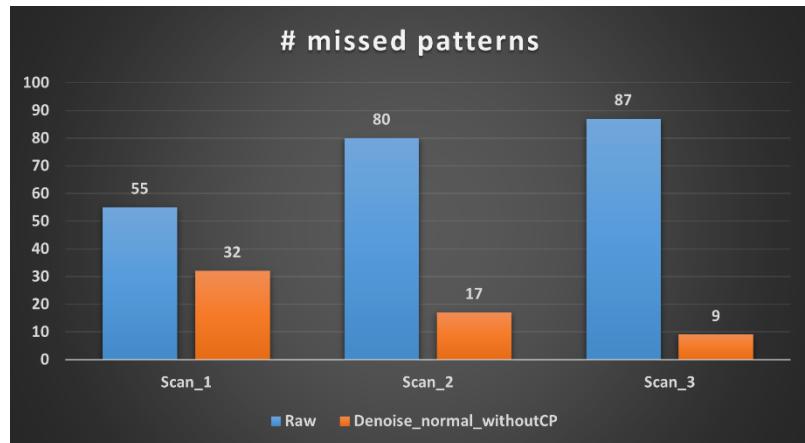
(c)

**Figure 5.13.** Plot of “# patterns with properly extracted contours” metric for noisy vs. denoised images with 3 different scan types (a) Scan\_1, (b) Scan\_2, (c) Scan\_3 on the Pillar patterns. The # patterns with fully extracted contours in denoise images are significantly higher than the ones in noise images across all the scanning modes for four different experimental blind-spot pixel manipulation mechanisms [ $p1, p2, p3, p4$ ]. Reprinted with permission from [76].

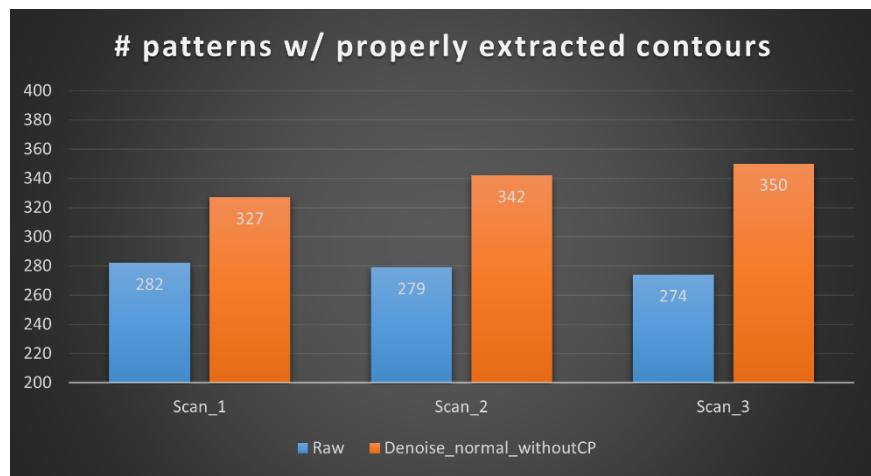
**Table 5.3.** Summary of analysis metric from SEMSuite<sup>TM</sup> (**Pillar** patterns). Reprinted with permission from [76].

Scanning direction	Images (Noisy/Denoised)	Gauge ID	Lay out patterns	# Patterns w/ properly extracted contours	# Patterns w/ partially extracted contours	# missed patterns
Scan_1	Raw	1	361	282	24	55
	Denoised_uniform_withCP	2		319	2	40
	Denoised_normal_withoutCP	3		327	2	32
	Denoised_normal_fitted	4		316	2	43
	Denoised_normal_additive	5		326	6	29
Scan_2	Raw	1	361	279	2	80
	Denoised_uniform_withCP	2		348	0	13
	Denoised_normal_withoutCP	3		342	2	17
	Denoised_normal_fitted	4		353	1	7

	Denoised_normal_additive	5		352	1	8
Scan_3	Raw	1	361	274	0	87
	Denoised_uniform_withCP	2		335	1	25
	Denoised_normal_withoutCP	3		350	2	9
	Denoised_normal_fitted	4		344	11	6
	Denoised_normal_additive	5		332	9	20



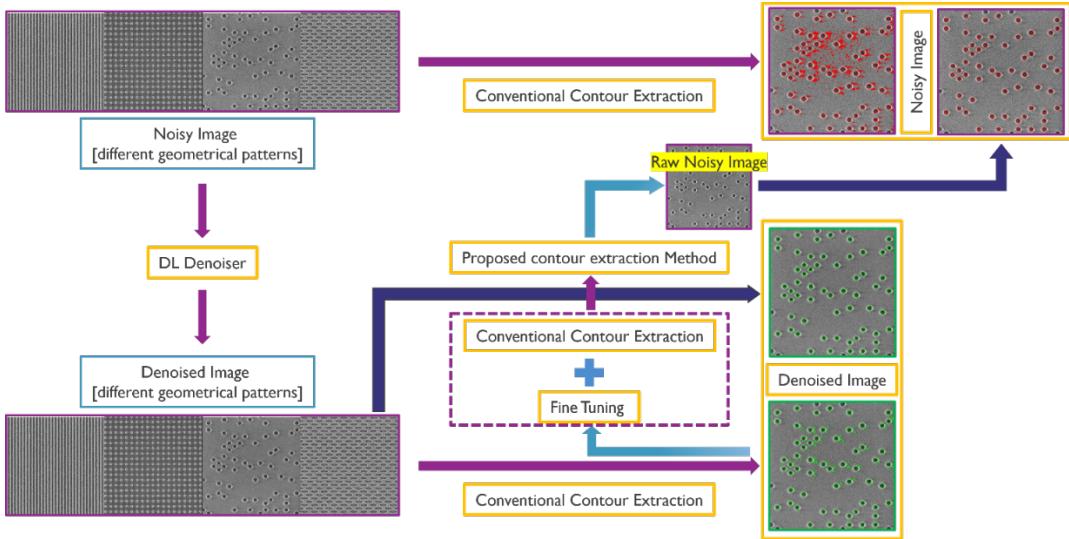
**Figure 5.14.** Plot of “# missed patterns” metric for noisy vs. de-noised images with 3 different scan types on the Pillar patterns. The # missed patterns in denoise images [only shown on p2] are significantly lower than the ones in noise images across all the scanning modes. Reprinted with permission from [76].



**Figure 5.15.** Plot of “# patterns with properly extracted contours” metric for noisy vs. de-noised images with 3 different scan types on the Pillar patterns. The # patterns with fully extracted contours in denoise images [only shown on p2] are significantly higher than the ones in noise images across all the scanning modes. Reprinted with permission from [76].

## **5.5. Proposed DL Denoiser Assisted Framework for Robust SEM Contour Extraction**

We have extended our work towards enabling SEM contour extraction possible for all the edges in the raw noisy SEM image itself. We have proposed a deep learning denoiser assisted framework for the extraction and analysis of SEM contours with main contributions as: (a) novel noise removal method, replacing conventional noise reduction techniques (as Gaussian/Median-blur etc.), with efficacy in edge extraction accuracy, (b) with minimum/no requirement of external user input or metadata (like *GDSII / OASIS* data, CSV-like meta-data set etc.) to extract and analyse information from noisy SEM images, (c) an improved contour extraction algorithm capable to extract contours on the body of noisy raw image itself with a posteriori knowledge derived from its denoised twins. The proposed contour extraction algorithm, aided by deep learning denoiser, specifically eliminates presence of any outlier patterns by analysing the shape properties such as #sides, perimeter, area etc., then differentiating from the target patterns. We have analysed, compared, and validated our contour extraction results for each noisy/denoised image pair for categorically different geometrical patterns such as L/S (line-space), C/H (contact-hole), pillars with different scan types, SRAM structure and Logic structure etc. We have demonstrated that our proposed method is capable to extract contours on the body of the noisy SEM images with accuracy in close proximity with design data. The proposed framework is shown in Fig. 5.16.

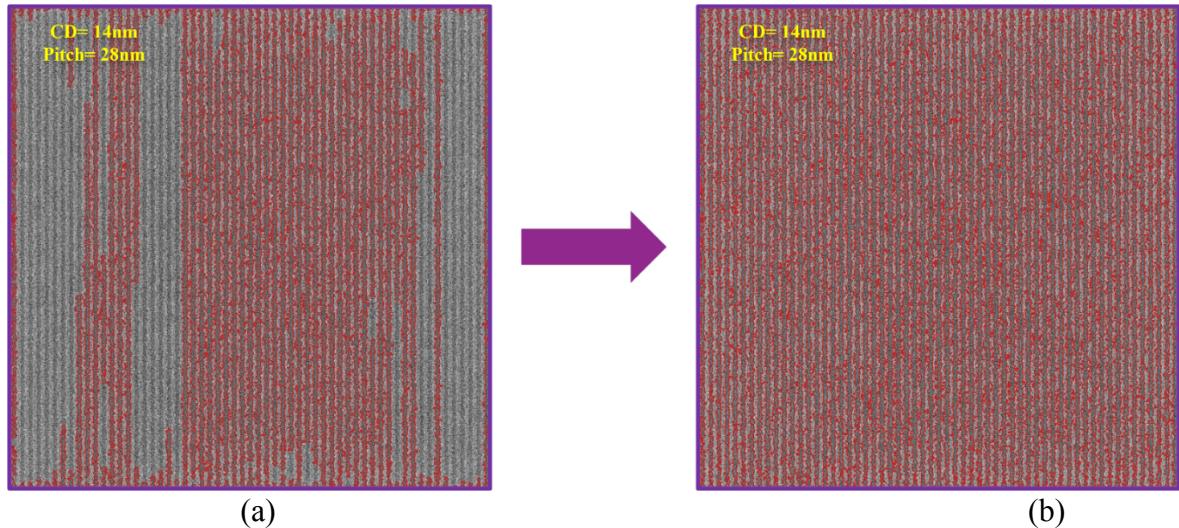


**Figure 5.16.** Proposed DL Denoiser assisted Framework.

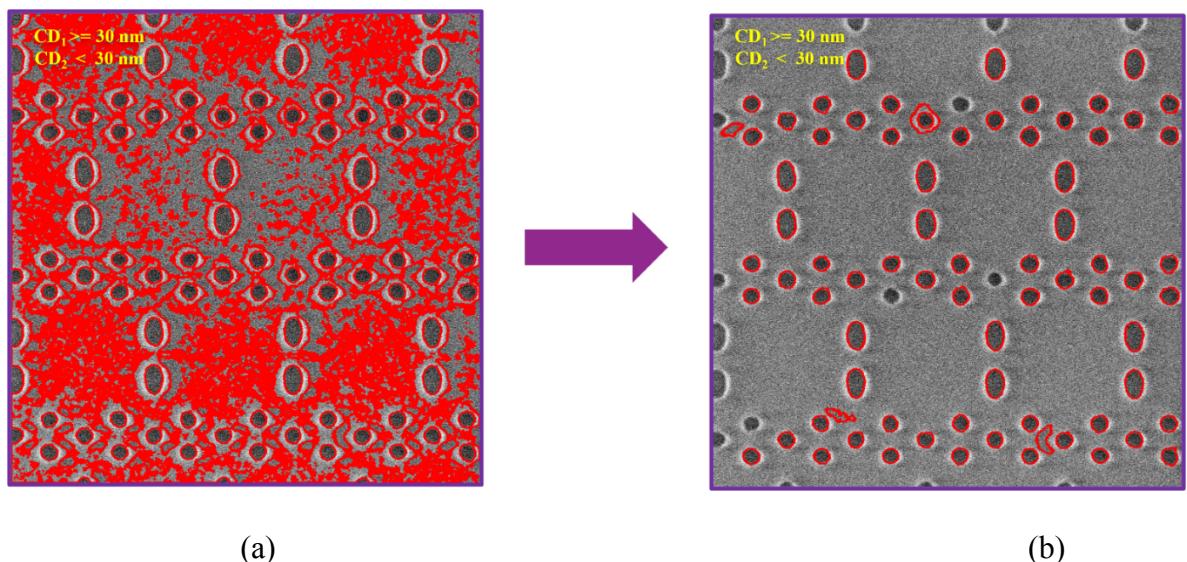
## 5.6. Experimental Results

Figs. [5.17, 5.18 and 5.19] (b) demonstrates comparison of extraction of contours on the body of the noisy images itself using our proposed framework, against using conventional image processing technique as shown in Figs. [5.17, 5.18 and 5.19] (a). With conventional method, for noisy images we are missing the granularity for major pixels and thus CD extraction is failing. For the same, proposed method makes better CD extraction possible. Our proposed approach is capable to extract contours on the body of the raw noisy image itself, where the contour extraction algorithm is only fine-tuned by the contour extraction parameters [based on shape analysis/contour analysis] of the denoised images. Varying parameters, for conventional method, has significant impact on the number of properly/partially extracted contours as well as missed patterns. However, our proposed contour extraction method shows robustness. It has also been observed for certain geometrical patterns, if contour extraction can be performed on noisy images without any image preprocessing feature, the contour extraction performance gain can be achieved using only its denoised twins approximately more than 100%. This is a significant metric to evaluate the performance of the proposed framework as how accurately we can extract

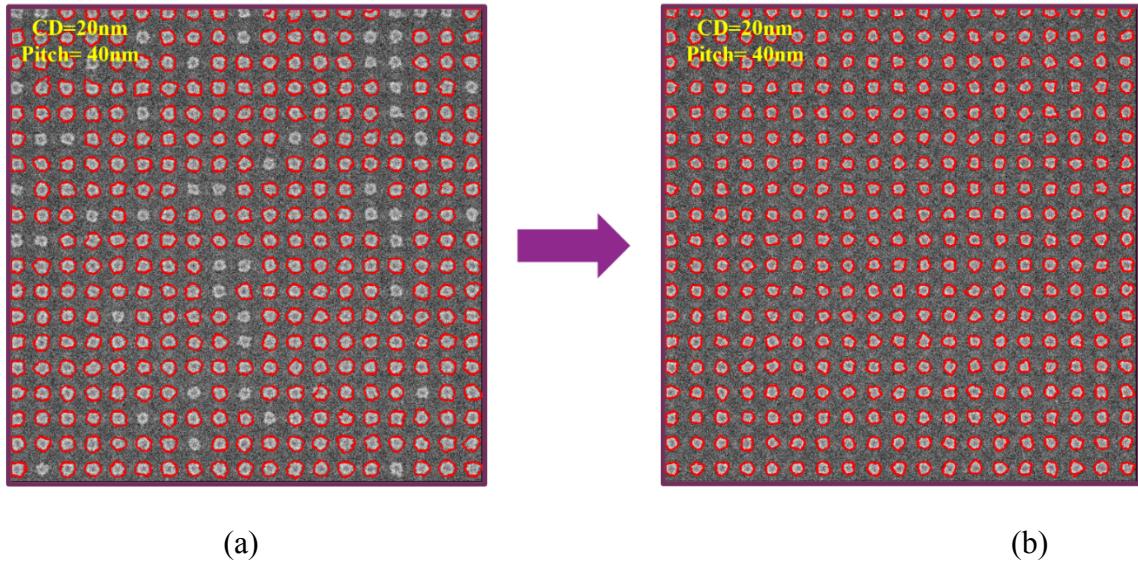
contours without using any additional image preprocessing steps. As shown in Table 5.4, for pillar patterns we were able to extract all contours without any “# missed patterns” on both raw noisy and denoised images.



**Figure 5.17.** L/S pattern [P28\_8nm] (ADI) thin resist: (a) Contour extraction performed using conventional image processing technique on noisy raw CD-SEM image, (b) Contour extraction performed using proposed method on the same noisy raw CD-SEM image.



**Figure 5.18.** SRAM structure [P28] (ADI): (a) Contour extraction performed using conventional image processing technique on noisy raw CD-SEM image, (b) Contour extraction performed using proposed method on the same noisy raw CD-SEM image.



**Figure 5.19.** Orthogonal pillar array (design CD=20nm, pitch= 40nm) with Scan\_1 type: (a) Contour extraction performed using conventional image processing technique on noisy raw CD-SEM image, (b) Contour extraction performed using proposed method on the same noisy raw CD-SEM image.

**Table 5.4.** Summary of analysis metric from our proposed framework in comparison with SEMSuite<sup>TM</sup> (Pillar patterns).

Scanning Direction	Images (Noisy/Denoised)	Noising ID	Method	Extracted contours/Layout Patterns	Error Percentage (%)	Method	Extracted contours/Layout Patterns	Error Percentage (%)
Scan_1	Raw	1	Conventional	310/361	14.13	Proposed Method	361/361	0
	Denoised_uniform_withCP	2		349/361	3.32		361/361	0
	Denoised_normal_withoutCP	3		319/361	11.63		361/361	0
	Denoised_normal_fitted	4		327/361	9.42		361/361	0
	Denoised_normal_additive	5		335/361	7.2		361/361	0
Scan_2	Raw	1	Conventional	283/361	21.6	Proposed Method	361/361	0
	Denoised_uniform_withCP	2		348/361	3.6		361/361	0
	Denoised_normal_withoutCP	3		344/361	4.7		361/361	0
	Denoised_normal_fitted	4		354/361	1.94		361/361	0
	Denoised_normal_additive	5		353/361	2.22		361/361	0
Scan_3	Raw	1	Conventional	275/361	23.82	Proposed Method	361/361	0
	Denoised_uniform_withCP	2		335/361	7.2		361/361	0
	Denoised_normal_withoutCP	3		349/361	3.32		361/361	0
	Denoised_normal_fitted	4		345/361	4.43		361/361	0
	Denoised_normal_additive	5		336/361	6.9		361/361	0

## 5.7. Summary

The main focus of this chapter is to demonstrate how the applied deep learning based denoising algorithm is helping to improve the contour detection/extraction capability/accuracy with after development (ADI) SEM images. Using a programmable tool (SEMSuite<sup>TM</sup>), we have analysed, compared, and validated our contour extraction results for each noisy/denoised image pair for categorically different geometrical patterns such as L/S (line-space), C/H (contact-hole), pillars with different scan types etc. We have also explored, different neighbourhood pixels manipulation strategies towards creating

blind-spot pixels and the corresponding denoising effect on the contour extraction performance. The comparative analysis demonstrates that de-noised images have higher confidence contour metric than their noisy twins while keeping the same parameter settings for both data input. Also, number of missed patterns and number of false bridges are significantly reduced with denoised images. Most importantly, this work shows with applied deep learning based denoising algorithm, the stochastic noise has been removed which in turn optimizing the varying non-uniform background intensity for better contour detection. Therefore, our proposed deep learning-based robust de-noising method facilitates improved contour detection/extraction capability to further provide more accurate metrology data (as estimation of CD/Pitch/resist sidewall/OVL/LER/LWR etc.) for OPC calibration as well as allows identification of defects with improved certainty. Finally, we have proposed an improved contour extraction algorithm capable to extract contours on the body of noisy raw image itself with a posteriori knowledge derived from its denoised twins. We have demonstrated how deep learning based denoising framework allows improved contour extraction with thin resists to be able to extract repeatable and accurate CDSEM metrology information for high NA EUV.

## **Chapter 6: Conclusion and Future Work**

Moore's prediction, which has been challenged at each critical technical bottleneck interface, has revived through innovative (a) node scaling strategies, (b) new materials search and transistor architectures itself, (c) advanced patterning strategies and (d) integration technologies. Lithography platforms also supported this innovation trend, as to enable constant pitch scaling, wavelength also shrunk from  $248\text{ nm}$  to  $193i$  to EUV. Today,  $5\text{ nm}$  chips are made with EUV (0.33 NA), but this has reached resolution limit around  $26\text{ nm}$ . Once again, technical innovation in form of ASML proposed high-NA (0.55 NA) will clear the major roadblocks by allowing print below  $26\text{ nm}$ , targeting HVM at around  $1\text{ nm}$  node. Therefore, to keep Moore's Law alive, we need to keep scaling alive through convergence of multi-disciplinary research and innovation. In this thesis, we have examined few key problems related to high-NA EUV lithography in the context aggressive pitches and thin resists and proposed apparent solutions by convergence of knowledge of artificial intelligence and metrology.

In the second chapter of this thesis, we discussed the constraints associated with existing process window monitoring methods. Conventional analysis of printable process window and defect-related process window features always requires manual comparison of CD-SEM images and therefore, involves a labour intensive and time-consuming effort. Therefore, the determination of these process limits is primarily based only on CD measurement and not completely on pattern fidelity. We proposed an alternative approach based on an unsupervised learning for determining process limits in an automated way against above stated problem. Our proposed deep learning-based framework, by learning the characteristic features from numerous CD-SEM images, provides a search ranking among the images based on a latent feature space distant metric, given a carefully chosen query image. This method appears to be more relevant for complex 2D shapes/patterns

where only CD- measurement is insufficient. In the future, our motivation is to extend this work towards more challenging random logic process window estimation.

In the third chapter of this thesis, we have discussed importance of SEM images for metrology and inspection and key challenges associated with its intrinsic noisy characteristic. Additionally, for high-NA EUV applicability, thin resist SEM images pose significant problem to extract repeatable and accurate metrology. We have discussed and analysed various state-of-the-art noise-removal techniques, both machine-learning and non-machine learning based, and emphasized their merits and drawbacks. We have proposed an U-Net architecture based unsupervised deep learning approach for denoising without requirement of any ground-truth or synthetic noiseless images. Against deep learning model convention, which require abundant dataset for training, our proposed method works effectively with very few or even a single noisy image. We have demonstrated experimentally that post denoising PSD in the lower frequency had not altered, which in turn indicates that the information content of the denoised images was not degraded by our proposed denoising approach, against a major concern highlighted by most prior approaches. Additionally, we have shown how our proposed approach helps to robustly detect pattern edges from the denoised images as well as to better identify defects. In chapter four and five, we have extended this work towards solving challenging defect inspection problems and robust contour detection and extraction for aggressive pitches and thin resists (High NA applications). In the future, we would like to improve the denoising algorithm by estimating a noise model precisely from a single noisy SEM image for different materials and process characterization.

In the fourth chapter of this thesis, we have proposed a novel robust supervised deep learning training scheme to accurately classify and localize different challenging defect types in SEM images with high degree of accuracy. We have demonstrated that an

ensemble framework based on a selective permutation of backbone architectures and an experimental selection of preference-based ensemble strategy to combine the output predictions from different models is effective for achieving enhanced performance on classification and detection task, against individual backbone. Our proposed model demonstrated stable performance in detecting defects with better accuracy for both noisy/denoised image pair and outperforms against existing conventional approaches, which are generally restricted by the manual selection of the detection threshold parameter as well as influenced by the image settings as noisy/denoised, resolution and field-of-view. Finally, a web-based defect inspection app has been built to enable different industry partners/vendors to run the application on their local servers/workstations on their own tool data. This will help to reduce engineering time as well as tool cycle time against manual inspection method associated with defect inspection process. We have extended this research further towards improved defect instance segmentation task along with previous classification and detection strategy by implementing Mask-RCNN based approach. By generating a precise extent of mask for each defect category/instance, our proposed method enabled to extract and calibrate each segmented mask and quantify the pixels that make up each mask, which in turn enabled us to count each categorical defect instances as well as to calculate the surface area in terms of pixels. These extracted parameters as area/length/width/defect coordinates are effective for potential analysis of root cause of these defects. Exhaustive data annotation has emerged as the major drawback of this proposed method. In the future, we would like to implement an unsupervised defect detection approach.

In the final chapter of this thesis, we have demonstrated how aforementioned deep learning denoising method enabled to improve the contour detection/extraction capability with thin resists by reducing the dependency of SEM image acquisition settings and to provide more

accurate metrology data for OPC calibration. We have compared and validated our contour extraction results, for each noisy/denoised image pair, for categorically different geometrical patterns using *SEMSuite<sup>TM</sup>*. The comparative analysis demonstrated that (a) missed patterns metric is significantly lower and (b) patterns with fully extracted contours metric is significantly higher in denoised images, against the ones in noisy images for all patterns. The number of false defects is also significantly reduced with denoised images. Finally, we have extended this research towards enabling SEM contour extraction possible for all the edges in the original noisy SEM image itself utilizing derived posteriori knowledge from its denoised twins. An improved contour extraction algorithm has been proposed to analyze the polygon parameters related to different geometrical patterns, such as shape, area, perimeter, length, width etc. to exclude outlier patterns and finally to return contours for intended patterns only. We have demonstrated that our proposed method is capable to extract contours on the body of the noisy SEM images with accuracy in close proximity with design data. In future, we would like to integrate an effective CNN based automated contour detection and extraction method.

Deep learning-based solutions possibly will play a pivotal role to advance Moore's law beyond 2 nm technology generations by convergence of hardware, software, and data. We look forward to it.

## Bibliography

- [1] G. E. Moore, "Progress in digital integrated electronics [Technical literature, Copyright 1975 IEEE. Reprinted, with permission. Technical Digest. International Electron Devices Meeting, IEEE, 1975, pp. 11-13.]," in IEEE Solid-State Circuits Society Newsletter, vol. 11, no. 3, pp. 36-37, Sept. 2006, doi: 10.1109/N-SSC.2006.4804410.
- [2] G. E. Moore, "Cramming more components onto integrated circuits, Reprinted from Electronics, volume 38, number 8, April 19, 1965, pp.114 ff.," in IEEE Solid-State Circuits Society Newsletter, vol. 11, no. 3, pp. 33-35, Sept. 2006, doi: 10.1109/N-SSC.2006.4785860.
- [3] <https://download.intel.com/newsroom/2022/manufacturing/id-transistor-graphic.jpg>.
- [4] Ryoung-han Kim, Yasser Sherazi, Peter Debacker, Praveen Raghavan, Julien Ryckaert, Arindam Malik, Diederik Verkest, Jae Uk Lee, Werner Gillijns, Ling Ee Tan, Victor Blanco, Kurt Ronse, Greg McIntyre, "IMEC N7, N5 and beyond: DTCO, STCO and EUV insertion strategy to maintain affordable scaling trend," Proc. SPIE 10588, Design-Process-Technology Co-optimization for Manufacturability XII, 105880N (20 March 2018); <https://doi.org/10.1117/12.2299335>.
- [5] <https://www.screen.co.jp/spe/en/products/300/dt-3000>.
- [6] <https://www.asml.com/en/products/duv-lithography-systems/twinscan-xt-860m>.
- [7] Mark Mueller, Serguei Komarov, Ki-Ho Baik, "Dry etching of chrome for photomasks for 100-nm technology using chemically amplified resist," Proc. SPIE 4754, Photomask and Next-Generation Lithography Mask Technology IX, (1 August 2002); <https://doi.org/10.1117/12.476954>.
- [8] Lord Rayleigh F.R.S. (1879) XXXI. Investigations in optics, with special reference to the spectroscope, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 8:49, 261-274, DOI: 10.1080/14786447908639684.
- [9] <https://www.edn.com/multi-patterning-strategies-for-navigating-the-sub-5-nm-frontier-part-1/>.
- [10] [http://www.lithoguru.com/scientist/litho\\_history/Lithographic\\_lens\\_history\\_and\\_evolution\\_Matsuyama\\_2006.pdf](http://www.lithoguru.com/scientist/litho_history/Lithographic_lens_history_and_evolution_Matsuyama_2006.pdf).
- [11] Bisschop, Peter De. "Optical proximity correction: A cross road of data flows." Japanese Journal of Applied Physics 55 (2016): n. pag.

- [12] Goodfellow, I., Bengio, Y., and Courville, A., [Deep Learning], The MIT Press (2016).
- [13] Halder, S., Leray, P., Sah, K., Cross, A., and Parisi, P., “Connected component analysis of review-SEM images for sub-10nm node process verification,” in [Metrology, Inspection, and Process Control for Microlithography XXXI], Sanchez, M. I., ed., 10145, 604 – 610, International Society for Optics and Photonics, SPIE (2017).
- [14] Halder, S., Cerbu, D., Saib, M., and Leray, P., “Review-sem image analysis with k-means algorithm: Am: Advanced metrology/di: Defect inspection,” in [2018 29th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC)], 255–258 (April 2018).
- [15] Leray, P., Halder, S., Lorusso, G., Baudemprez, B., Inoue, O., and Okagawa, Y., “Hybrid overlay metrology for high order correction by using CDSEM,” in [Metrology, Inspection, and Process Control for Microlithography XXX], Sanchez, M. I., ed., 9778, 691 – 698, International Society for Optics and Photonics, SPIE (2016).
- [16] Bisschop, P. D. and Hendrickx, E., “Stochastic printing failures in EUV lithography,” in [Extreme Ultraviolet (EUV) Lithography X], Goldberg, K. A., ed., 10957, 37 – 56, International Society for Optics and Photonics, SPIE (2019).
- [17] Bisschop, P. D., “Stochastic effects in EUV lithography: random, local CD variability, and printing failures,” Journal of Micro/Nanolithography, MEMS, and MOEMS 16(4), 1 – 17 (2017).
- [18] Halder, S., Mailfert, J., Leray, P., Rio, D., Peng, Y.-H., and Laenens, B., “Design-based metrology: beyond CD/EPE metrics to evaluate printability performance,” in [Metrology, Inspection, and Process Control for Microlithography XXX], Sanchez, M. I., ed., 9778, 290 – 300, International Society for Optics and Photonics, SPIE (2016).
- [19] Leray, P., Halder, S., Lorenzo, P. D., Wang, F., Zhang, P., Fang, W., Liu, K., and Jau, J., “Study of design-based e-beam defect inspection for hotspot detection and process window characterization on 10nm logic device,” in [Metrology, Inspection, and Process Control for Microlithography XXX], Sanchez, M. I., ed., 9778, 234 – 240, International Society for Optics and Photonics, SPIE (2016).
- [20] Ward, B. S., Berthiaume, S., Brist, T., and Brooker, P., “Metrology data cleaning and statistical assessment flow for modeling applications,” in [Metrology, Inspection, and Process Control for Microlithography XXIV], Raymond, C. J., ed., 7638, 646 – 656, International Society for Optics and Photonics, SPIE (2010).
- [21] LeCun, Y., Bengio, Y., et al., “Convolutional networks for images, speech, and time series,” The handbook of brain theory and neural networks 3361(10), 1995 (1995).

- [22] Karpathy, A., “Stanford university cs231n: Convolutional neural networks for visual recognition,” URL: <http://cs231n.stanford.edu/syllabus.html> (2018).
- [23] LeCun, Y., Bengio, Y., and Hinton, G., “Deep learning,” *nature* 521(7553), 436 (2015).
- [24] Krizhevsky, A., Sutskever, I., and Hinton, G. E., “Imagenet classification with deep convolutional neural networks,” in [Advances in neural information processing systems], 1097–1105 (2012).
- [25] Tschannen, M., Bachem, O., and Lucic, M., “Recent advances in autoencoder-based representation learning,” (2018).
- [26] Kingma, D. P. and Welling, M., “An introduction to variational autoencoders,” (2019).
- [27] Chollet, F. et al., “Keras: Deep learning library for theano and tensorflow,” URL: <https://keras.io/> 7(8), T1 (2015).
- [28] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al., “Tensorflow: A system for large-scale machine learning,” in [12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)], 265–283 (2016).
- [29] Ioffe, S. and Szegedy, C., “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” arXiv preprint arXiv:1502.03167 (2015).
- [30] Bottou, L., “Large-scale machine learning with stochastic gradient descent,” in [Proceedings of COMPSTAT’2010], 177–186, Springer (2010).
- [31] Kingma, D. P. and Ba, J., “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980 (2014).
- [32] Guo, G., Wang, H., Bell, D., Bi, Y., and Greer, K., “Knn model-based approach in classification,” in [OTM Confederated International Conferences” On the Move to Meaningful Internet Systems”], 986–996, Springer (2003).
- [33] Bappaditya Dey, Dorin Cerbu, Kasem Khalil, Sandip Halder, Philippe Leray, Sayantan Das, Yasser Sherazi, Magdy A. Bayoumi, Ryoung Han Kim, “Unsupervised machine learning based CD-SEM image segregator for OPC and process window estimation,” Proc. SPIE 11328, Design-Process-Technology Co-optimization for Manufacturability XIV, 113281G (23 March 2020); <https://doi.org/10.1117/12.2552055>.

- [34] Marturi, N., Demb'el'e, S., and Piat, N., "Scanning electron microscope image signal-to-noise ratio monitoring for micro-nanomanipulation," *Scanning* 36(4), 419–429 (2014).
- [35] Huang, C.-W. and Narayanan, S. S., "Deep convolutional recurrent neural network with attention mechanism for robust speech emotion recognition," in [2017 IEEE international conference on multimedia and expo (ICME)], 583–588, IEEE (2017).
- [36] Khalil, K., Eldash, O., Kumar, A., and Bayoumi, M., "Machine learning-based approach for hardware faults prediction," *IEEE Transactions on Circuits and Systems I: Regular Papers* 67(11), 3880–3892 (2020).
- [37] Ke, Y. and Hagiwara, M., "A natural language processing neural network comprehending english," in [2015 International Joint Conference on Neural Networks (IJCNN)], 1–7, IEEE (2015).
- [38] Khalil, K., Eldash, O., Kumar, A., and Bayoumi, M., "Intelligent fault-prediction assisted self-healing for embryonic hardware," *IEEE Transactions on Biomedical Circuits and Systems* 14(4), 852–866 (2020).
- [39] Shi, B., Bai, X., and Yao, C., "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE transactions on pattern analysis and machine intelligence* 39(11), 2298–2304 (2016).
- [40] Khalil, K., Eldash, O., Dey, B., Kumar, A., and Bayoumi, M., "A novel reconfigurable hardware architecture of neural network," in [2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS)], 618–621 (2019).
- [41] Anthony, M. and Bartlett, P. L., [Neural network learning: Theoretical foundations], cambridge university press (2009).
- [42] Khalil, K., Eldash, O., Dey, B., Kumar, A., and Bayoumi, M., "Architecture of a novel low-cost hardware neural network," in [2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS)], 1060–1063 (2020).
- [43] Wang, L., [Support vector machines: theory and applications], vol. 177, Springer Science & Business Media (2005).
- [44] Khalil, K., Eldash, O., Kumar, A., and Bayoumi, M., "Economic LSTM approach for recurrent neural networks," *IEEE Transactions on Circuits and Systems II: Express Briefs* 66(11), 1885–1889 (2019).
- [45] Sinaga, K. P. and Yang, M.-S., "Unsupervised k-means clustering algorithm," *IEEE Access* 8, 80716–80727 (2020).

- [46] Caron, M., Bojanowski, P., Joulin, A., and Douze, M., “Deep clustering for unsupervised learning of visual features,” in [Proceedings of the European Conference on Computer Vision (ECCV)], 132–149 (2018).
- [47] Yu, L., Zhou,W., Pu, L., and Fang,W., “SEM image quality enhancement: an unsupervised deep learning approach,” in [Metrology, Inspection, and Process Control for Microlithography XXXIV], Adan, O. and Robinson, J. C., eds., 11325, 388 – 396, International Society for Optics and Photonics, SPIE (2020).
- [48] Midoh, Y. and Nakamae, K., “Image quality enhancement of a CD-SEM image using conditional generative adversarial networks,” in [Metrology, Inspection, and Process Control for Microlithography XXXIII], Ukrantsev, V. A. and Adan, O., eds., 10959, 37 – 46, International Society for Optics and Photonics, SPIE (2019).
- [49] Cerbu, D., Halder, S., and Leray, P., “Deep-learning-based SEM image denoiser (Conference Presentation),” in [Metrology, Inspection, and Process Control for Microlithography XXXIII], Ukrantsev, V. A. and Adan, O., eds., 10959, International Society for Optics and Photonics, SPIE (2019).
- [50] Chaudhary, N., Savari, S. A., and Yeddulapalli, S. S., “Line roughness estimation and Poisson denoising in scanning electron microscope images using deep learning,” *Journal of Micro/Nanolithography, MEMS, and MOEMS* 18(2), 1– 16 (2019).
- [51] Krull, A., Buchholz, T.-O., and Jug, F., “Noise2void-learning denoising from single noisy images,” in [Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition], 2129–2137 (2019).
- [52] Ronneberger, O., Fischer, P., and Brox, T., “U-net: Convolutional networks for biomedical image segmentation,” in [Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015], Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F., eds., 234–241, Springer International Publishing, Cham (2015).
- [53] Mack, C. A. and Bunday, B. D., “Analytical linescan model for SEM metrology,” in [Metrology, Inspection, and Process Control for Microlithography XXIX], Cain, J. P. and Sanchez, M. I., eds., 9424, 117 – 139, International Society for Optics and Photonics, SPIE (2015).
- [54] Søren Dramsch, J., Amour, F., and Lüthje, M., “Gaussian mixture models for robust unsupervised scanning-electron microscopy image segmentation of north sea chalk,” in [Proceedings of the First EAGE/PESGBWorkshop on Machine Learning (London2018)], 28–30, European Association of Geoscientists and Engineers (2018). First EAGE/PESGB Workshop Machine Learning; Conference date: 29-11-2018 Through 30-11-2018.

- [55] Lindsay, B. G., "Mixture models: theory, geometry and applications," in [NSF-CBMS regional conference series in probability and statistics], i–163, JSTOR (1995).
- [56] Bappaditya Dey, Sandip Halder, Kasem Khalil, Gian Lorusso, Joren Severi, Philippe Leray, Magdy A. Bayoumi, "SEM image denoising with unsupervised machine learning for better defect inspection and metrology," Proc. SPIE 11611, Metrology, Inspection, and Process Control for Semiconductor Manufacturing XXXV, 1161115 (22 February 2021); <https://doi.org/10.1117/12.2584803>.
- [57] J. Kim, S. Kim, N. Kwon, H. Kang, Y. Kim, and C. Lee, "Deep learning based automatic defect classification in through-silicon via process: Fa: Factory automation," in 2018 29th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC), 2018, pp. 35–39. [Online]. Available: <http://dx.doi.org/10.1109/ASMC.2018.8373144>.
- [58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.
- [59] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: <http://arxiv.org/abs/1409.1556>.
- [60] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," vol. 313, no. 5786, pp. 504–507, 2006. [Online]. Available: <https://doi.org/10.1126/science.1127647>.
- [61] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in Computer Vision – ECCV 2016, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 21–37.
- [62] F. Timischl, M. Date, and S. Nemoto, "A statistical model of signal–noise in scanning electron microscopy," Scanning, vol. 34, no. 3, pp. 137–144, 2012.
- [63] R. Zwitch, "Streamlit," <https://docs.streamlit.io/en/stable/>.
- [64] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998. [Online]. Available: <https://doi.org/10.1109/5.726791>.

- [65] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1–9. [Online]. Available: <https://doi.org/10.1109/CVPR.2015.7298594>.
- [66] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2999–3007. [Online]. Available: <https://doi.org/10.1109/ICCV.2017.324>.
- [67] M. Sharifzadeh, S. Alirezaee, R. Amirkhattabi, and S. Sadri, “Detection of steel defect using the image processing algorithms,” in 2008 IEEE International Multitopic Conference, 2008, pp. 125–127. [Online]. Available: <https://doi.org/10.1109/INMIC.2008.4777721>.
- [68] S. Kim and I.-S. Oh, “Automatic defect detection from sem images of wafers using component tree,” Journal of Semiconductor Technology and Science, vol. 17, pp. 86–93, 2017.
- [69] R. Bonam, S. Halle, D. Corliss, H.-Y. Tien, F. Wang, W. Fang, and J. Jau, “E-beam inspection of euv programmed defect wafers for printability analysis,” in ASMC 2013 SEMI Advanced Semiconductor Manufacturing Conference, 2013, pp. 310–314. [Online]. Available: <https://doi.org/10.1109/ASMC.2013.6552751>.
- [70] J. Wang, Z. Yang, J. Zhang, Q. Zhang, and W.-T. K. Chien, “Adabalgan: An improved generative adversarial network with imbalanced learning for wafer defective pattern recognition,” IEEE Transactions on Semiconductor Manufacturing, vol. 32, no. 3, pp. 310–319, 2019. [Online]. Available: <https://doi.org/10.1109/TSM.2019.2925361>.
- [71] J.-H. Lee and J.-H. Lee, “A reliable defect detection method for patterned wafer image using convolutional neural networks with the transfer learning,” IOP Conference Series: Materials Science and Engineering, vol. 647, p. 012010, oct 2019. [Online]. Available: <https://doi.org/10.1088/1757-899x/647/1/012010>.
- [72] B. Devika and N. George, “Convolutional neural network for semiconductor wafer defect detection,” in 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019, pp. 1–6. [Online]. Available: <https://doi.org/10.1109/ICCCNT45670.2019.8944584>.
- [73] J.-C. Chien, M.-T. Wu, and J.-D. Lee, “Inspection and classification of semiconductor wafer surface defects using cnn deep learning networks,” Applied Sciences, vol. 10, no. 15, 2020. [Online]. Available: <https://www.mdpi.com/2076-3417/10/15/5340>.
- [74] Y. Yuan-Fu, “A deep learning model for identification of defect patterns in semiconductor wafer map,” in 2019 30th Annual SEMI Advanced Semiconductor

Manufacturing Conference (ASMC), 2019, pp. 1–6. [Online]. Available: <https://doi.org/10.1109/ASMC.2019.8791815>.

- [75] D. V. Patel, R. K. Bonam, and A. A. Oberai, “Deep learning-based detection, classification, and localization of defects in semiconductor processes,” Journal of Micro/Nanolithography, MEMS, and MOEMS, vol. 19, no. 2, pp. 1 – 15, 2020. [Online]. Available: <https://doi.org/10.1117/1.JMM.19.2.024801>.
- [76] B. Dey, S. Wu, S. Das, K. Khalil, S. Halder, P. Leray, S. Bhamidipati, K. Ahi, M. Pereira, G. Fenger, and M. Bayoumi, “Unsupervised machine learning based sem image denoising for robust contour detection,” in International Conference on Extreme Ultraviolet Lithography 2021, vol. 11854. SPIE, 2021, pp. 44–58.
- [77] T.-Y. Lin, P. Doll’ar, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” 2017.
- [78] A. Casado-Garc’ia and J. Heras, “Ensemble methods for object detection,” 2019, <https://github.com/ancasag/ensembleObjectDetection>.
- [79] I. Lütkebohle, “LabelImg,” <https://github.com/tzutalin/labelImg>, 2015.
- [80] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, “Generalized intersection over union: A metric and a loss for bounding box regression,” in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 658–666. [Online]. Available: <https://doi.org/10.1109/CVPR.2019.00075>.
- [81] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2980-2988, doi: 10.1109/ICCV.2017.322.
- [82] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, “Detectron2,” <https://github.com/facebookresearch/detectron2>, 2019.
- [83] <https://github.com/fizyr/keras-retinanet>.
- [84] E. Dehaerne, B. Dey, and S. Halder, “Ensemble deep learning-based defect classification and detection in sem images.” [Online]. Available: Ensemble Deep Learning-based Defect Classification and Detection in SEM Images ([learnopencv.com](http://learnopencv.com)).
- [85] Kawada, H., Ebizuka, Y., Sutani, T., and Kawasaki, T., “LER and LWR measurements used for monitoring wiggling and stochastic-failure (Conference Presentation),” in [Metrology, Inspection, and Process Control for Microlithography XXXIII], Ukrantsev, V. A. and Adan, O., eds., 10959, International Society for Optics and Photonics, SPIE (2019).

- [86] Weisbuch, F., Schatz, J., Mattick, S., Schuch, N., Figueiro, T., and Schiavone, P., “Investigating SEM-contour to CDSEM matching,” in [Metrology, Inspection, and Process Control for Semiconductor Manufacturing XXXV], Adan, O. and Robinson, J. C., eds., 11611, 167 – 181, International Society for Optics and Photonics, SPIE (2021).
- [87] Schneider, L., Farys, V., Serret, E., and Fenouillet-Beranger, C., “Framework for SEM contour analysis,” in [Metrology, Inspection, and Process Control for Microlithography XXXI], Sanchez, M. I., ed., 10145, 376 – 388, International Society for Optics and Photonics, SPIE (2017).
- [88] Shibahara, T., Minakawa, T., Oikawa, M., Shindo, H., Sugahara, H., and Hojyo, Y., “A CD-gap-free contour extraction technique for OPC model calibration,” in [Metrology, Inspection, and Process Control for Microlithography XXV], Raymond, C. J., ed., 7971, 217 – 224, International Society for Optics and Photonics, SPIE (2011).
- [89] Zhou, T., Shi, X., YanYan, Li, C., Chen, S., Zhao, Y., Zhou, W., Zhou, K., and Zeng, X., “An effective method of contour extraction for sem image based on dcnn,” in [2020 International Workshop on Advanced Patterning Solutions (IWAPS)], 1–4 (2020).
- [90] Schneider, C. A., Rasband, W. S., and Eliceiri, K. W., “Nih image to imagej: 25 years of image analysis,” Nature methods 9(7), 671–675 (2012).

Dey, Bappaditya. Bachelor of Science, University of Calcutta, India, Spring 2006; Master of Science, University of Calcutta, India, Spring 2008; Master of Technology, West Bengal University of Technology, India, Spring 2010; Master of Science, University of Louisiana at Lafayette, Fall 2017; Doctor of Philosophy, University of Louisiana at Lafayette, Spring 2022

Major: Computer Engineering

Title of Dissertation: Machine Learning Algorithms and Architectures for SEM Based Defect Inspection and Metrology Challenges in Semiconductor Manufacturing

Dissertation Director: Dr. Magdy A. Bayoumi

Pages in Dissertation: 153; Words in Abstract: 345

## **Abstract**

Moore's law is one of the key driving forces behind technology node scaling in semiconductor manufacturing process. Various design tricks and process tricks are being implemented to maintain PPAC scaling as we migrate from node to node. Fabrication of ICs, while adhering to node scaling roadmaps, emerged as a new challenge to the foundries/IDM's in many aspects. While reduced feature size of a transistor improved its switching-speed and power efficiency, it demands more complex and often new process technology, more capital and research expertise. As EUV based lithography gets adopted to keep scaling semiconductor devices in next-generation chips, new metrology and inspection challenges arise. We need to measure these small dimensions fast but also without sacrificing accuracy and repeatability. Metrology and Inspection are at the heart of process control. Without adequate metrology and inspection capability yields suffer. Therefore, as we continue to shrink the pitch (below 36 nm), measurement of feature dimensions as CD, Pitch, SWA, LER, LWR, LCDU etc. and classification of stochastic defects become a key issue. In this research, we have investigated and applied several deep learning algorithms and architectures for solving certain E-beam based inspection and metrology problems in the context of advanced node semiconductor manufacturing. The novelty of our proposed method includes: (i) an unsupervised machine learning based

approach for pre-processing OPC data and to automatically define defect process windows, (ii) an unsupervised deep learning denoiser assisted framework for the extraction and analysis of SEM contours with main contributions as: (a) a novel noise removal method, replacing conventional noise reduction techniques (as Gaussian/Median-blur etc.), with efficacy in edge extraction accuracy; (b) with minimum/no requirement of external user input or metadata (like GDSII / OASIS data, CSV-like meta-data set etc.) to extract and analyze information from noisy SEM images; (c) an improved contour extraction algorithm capable to extract contours on the body of noisy raw image itself with a posteriori knowledge derived from its denoised twins; (d) analysis and validation of our contour extraction results for different geometrical patterns using a programmable tool (『SEMSuite』 ^TM), (iii) finally, we have proposed an ADCDS framework for enhanced defect inspection.

### **Biographical Sketch**

Bappaditya Dey received his B.Sc. and M.Sc. degrees in physics (Hons.) and electronic science from University of Calcutta, Kolkata, India, in 2006 and 2008, respectively, and the second M. Tech degree in computer science and engineering from MAKAUT (formerly known as WBUT), Kolkata, India in 2010. He also received his third M.Sc. degree in computer engineering from the Center for Advanced Computer Studies (CACS), University of Louisiana at Lafayette, USA in 2017. He completed the requirements for the degree of Doctor of Philosophy in computer from the University of Louisiana at Lafayette in Spring, 2022. Bappaditya Dey was a Graduate Teaching and Research Assistant at University of Louisiana at Lafayette since August 2015. He joined imec, Belgium in 2018 as a R&D Engineer and worked in various roles since then. He is a member of IEEE and SPIE. His research interests include VLSI, reconfigurable hardware, machine learning, intelligent hardware systems, computer vision, artificial intelligence, and semiconductor process optimization.

ProQuest Number: 29209032

INFORMATION TO ALL USERS

The quality and completeness of this reproduction is dependent on the quality  
and completeness of the copy made available to ProQuest.



Distributed by ProQuest LLC (2023).

Copyright of the Dissertation is held by the Author unless otherwise noted.

This work may be used in accordance with the terms of the Creative Commons license  
or other rights statement, as indicated in the copyright statement or in the metadata  
associated with this work. Unless otherwise specified in the copyright statement  
or the metadata, all rights are reserved by the copyright holder.

This work is protected against unauthorized copying under Title 17,  
United States Code and other applicable copyright laws.

Microform Edition where available © ProQuest LLC. No reproduction or digitization  
of the Microform Edition is authorized without permission of ProQuest LLC.

ProQuest LLC  
789 East Eisenhower Parkway  
P.O. Box 1346  
Ann Arbor, MI 48106 - 1346 USA