Machine Learning Based SEM Image Analysis for Automatic Detection and Classification of Wafer Defects

Sanghyun Choi
Siemens EDA
South Korea
sh.choi@siemens.com

Qian Xie
Siemens EDA
USA
qian.xie@siemens.com

Nathan Greeneltch
Siemens EDA
USA
nathan.greeneltch@siemens.com

Hyung Joo Lee
Siemens EDA
South Korea
hyungjoo.lee@siemens.com

Mohan Govindaraj
Siemens EDA
USA
rangan.govindaraj@siemens.com

Srividya Jayaram
Siemens EDA
USA
srividya.jayaram@siemens.com

Mark Pereira
Siemens EDA
India
mark.pereira@siemens.com

Sayani Biswas
Siemens EDA
India
sayani.biswas@siemens.com

Samir Bhamidipati
Siemens EDA
India
bhamidipati.samir@siemens.com

Ilhami Torunoglu
Siemens EDA
USA
ilhami.torunoglu@siemens.com

***Abstract*—A machine learning (ML) based approach is proposed for analyzing Scanning Electron Microscope (SEM) images and classifying review defects. Accurate and timely SEM image analysis is crucial and impacts manufacturing yield. The state-of-the-art object detection model YOLOv8 (You Only Look Once version 8) is used as it offers a good balance between accuracy and inference speed. This work demonstrates the utility of YOLO for SEM ADC and extends capability using ensemble voting to achieve higher quality results.**

Keywords—ADC (automatic defect classification), wafer defects, SEM (scanning electron microscope), ML model, YOLO model

I. Introduction

In the current semiconductor fab process flow, defect analysis is done by human experts that can be both time consuming and prone to error as it requires long hours of focus. Software with hand-crafted rules to detect defects may be used to overcome the problem, but such rules are challenging to construct as image quality will usually vary for every layer-process combination. Using machine learning (ML), a model can learn complex rules and generalize to various image qualities, enabling accurate detection of defects without human intervention. Experimental results confirm that trained YOLOv8 (You Only Look Once version 8) [1], [2] model can predict 6 types of defects with a mean Average Precision (mAP) of 0.79 (at IoU=0.5, where IoU is Intersection Over Union – metric used to evaluate Deep Learning algorithms by estimating how well a predicted mask or bounding box matches the ground truth data) for unseen real-world fab process scanning electron microscope (SEM) images with varying image qualities. The training set is augmented with synthetic images containing relevant features and image attributes, and the prediction routine is extended into an ensemble approach to mimic automatic "ML recipe" creation. Special consideration was given to modern semiconductor fab resource constraints and the profiles of the potential users of the solution.

II. Methods

A. Model Description

Previous works [3]-[6] use YOLO to detect wafer defects in SEM images but the focus is on finding certain patterns in SEM images without comparing with intended design. Combining SEM and layout images as 2 channels was done in [7] but with the use of Generative Adversarial Network (GAN) model for defect detection involving only two defect types. Fig. 1 shows our current approach.

YOLOv8 model is trained to receive as input a multichannel image with 1) SEM image in first channel, and 2) aligned design layout clip in second channel, and to predict as outputs 1) defect locations (via bounding boxes) and 2) defect types. The input can be extended to a third channel, for instance with extracted contours, but is not utilized in the current work. Based on this training scheme, the model is expected to learn abnormalities in SEM images by using layout images as reference. The defect types the model is trained to predict are missing pattern (M), added pattern (A), pinch (P), line-end extension (LE), line-end

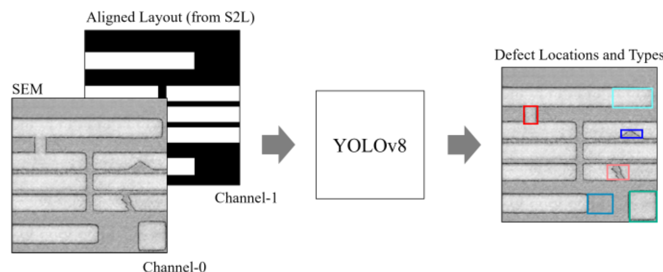

Fig. 1. YOLOv8 is trained to predict defect locations and defect types by comparing SEM image with aligned layout image.

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

pullback (LP), and bridge (B). Fig. 2 shows a sample prediction for each defect type. authors assert sufficient coverage for each type in this study.

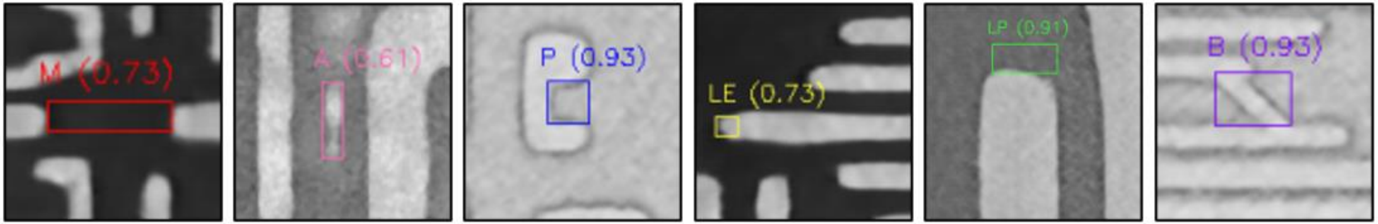

Fig. 2. Sample predictions for various defect types. From left to right: Missing Pattern (M), Added Pattern (A), Pinch (P), line-end extension (LE), line-end pullback (LP), and bridge (B).

B. Dataset

It is difficult to obtain sufficient SEM images and design layouts for model training as they are rarely shared by wafer fabs. Moreover, not every SEM image contains a defect type that the model is trained to predict. Therefore, a GAN-based data augmentation technique is employed (Fig. 3) to grow and have more control over the input set. Specifically, Conditional GAN (cGAN) [8] for image-to-image translation is used. First, using non-defective SEM/layout image pairs, the GAN is trained to generate synthetic SEM images from layout images. Then, by deliberately introducing defects into layout images, for example, by randomly erasing/adding patterns, and feeding in the modified layout image into trained generator model, synthetic SEM images with desired defects is obtained. To add realistic variety to the synthetic set, multiple cGAN models were trained to generate SEM images from different vendor sources, wafer fabs, and process conditions.

Although resource constraints provide a natural governor to the size of the training set, in general more data and more examples leads to a better model. The final dataset used for modeling consists of 355 actual images from 5 wafer fabs, and 1,302 synthetic images. 80% of actual images and 100% of synthetic images are used for training and the remaining for testing. The number of instances of each defect type is shown in Table I. While there persists an imbalance in defect counts, the authors assert sufficient coverage for each type in this study.

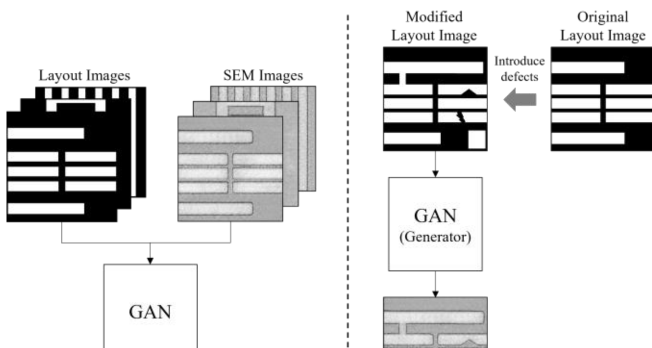

Fig. 3. GAN is trained to generate synthetic SEM image for a given layout image.

TABLE I
NUMBER OF INSTANCES FOR EACH DEFECT TYPE

Image Type	Defect Type					
	M	A	P	LE	LP	B
Synthetic	1,586	1,201	361	135	158	735
Actual	1,198	187	286	533	510	31
Total	2,784	1,388	647	668	668	766

C. Model Training

Depending on the choice of model architecture and the number of epochs, the model is trained on a Linux (RHEL8) machine equipped with two NVIDIA A100 GPU cards, training took 2~10 hours. Therefore, automated hyperparameter tuning is not considered in this work and many hyperparameters, along with train settings, were chosen heuristically based on trial-and-error. Every model is trained for 500 epochs with automatically determined batch size, 640x640 image size, and data augmentation with on-the-fly image transformations such as scale, shear, and flip.

III. Results

A. Single Model with Various Architectures

Five different YOLOv8 architectures were studied by varying the size of the feature extraction backbone of the network, which is the first block of a YOLOv8 forward pass. Fig. 4 shows the trained model performance on unseen test data, in terms of mean Average Precision (mAP) at IoU=0.5, when different architectures of YOLOv8 are chosen. The IoU threshold determines the required accuracy of the detected defect locations (bounding boxes). A value IoU=0.5 indicates that a predicted bounding box is correct if it overlaps at least 50% with the ground truth bounding box. The authors observe that performance spikes when using the 'Medium' model. Inference latency is affected by the size of the model and should be an important consideration for other studies. However, since the mAP performance difference is so stark, the tradeoff between speed and accuracy was not rigorously studied.

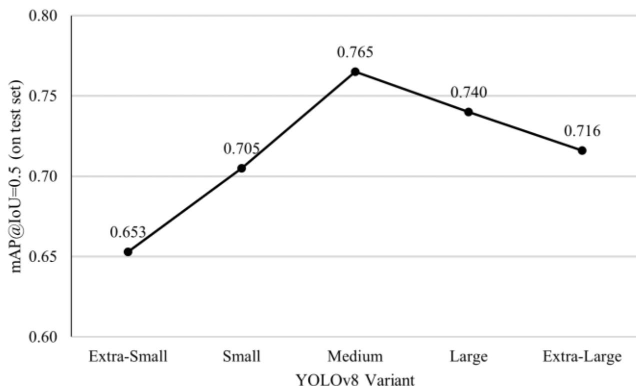

Fig. 4. Overall single model performance on test data depending on architecture shown in mAP(IoU=0.5).

B. Prediction Results by Defect Type

The best performing model was built with the "Medium" YOLOv8 architecture, and the prediction results for this model are included in Table II. The per-class performance shows a distinct difference in quality depending on defect type. This result suggests that a single model should not be used for all defect types in production. A more in depth look at defect-specific performance of each model architecture is shown in Table III.

A study of Table III reinforces the give-and-take nature of model performance in the SEM ADC application with best defect performance spreading across "Small", "Medium", and "Large" model architectures. While this is a concerning issue, it can be overcome by targeting different architectures to specific defect types to create an "ML recipe". Due to the lack of "ML recipe" intuition for fab engineers, any usage of ML SEM ADC modeling in production will require automated recipe setup. Towards this goal, the following section will discuss a potential solution to this issue.

TABLE II
MODEL PERFORMANCE ON TAIN/TEST DATA

Dataset	Per-class AP at IoU=0.5					
	M	A	P	LE	LP	B
Train	0.892	0.969	0879	.0779	0.830	0.985
Test	0.765	0.477	0.832	0.883	0.849	0.901

B. Ensemble Voting

The next step in this work is to investigate ways of turning the ML SEM ADC model into a manageable solution for fab engineers. Resource constraints, including time-to-recipe and limits to ML modeling knowledge in the fab, must be considered and addressed if this solution is to be adopted.

Table III shows the performance of best models with respect to individual defect types. 'Small' model shows best performance for detecting bridges in train set, while 'Medium' model shows best performance for detecting missing patterns, pinches, and line-end extensions/pullbacks in test set. Finally, 'Large' model shows best performance for detecting added patterns and bridges in test set. Based on this result, it is clear that a single model will not predict well on all defect types. It is also not ideal to have different models (i.e, "ML recipes") for each defect type. Therefore, the five models are combined for use in an ensemble voting prediction routine.

The ensemble voting strategy is shown in Fig. 5. All relevant models predict on an image and all resulting bounding boxes are collected. With the IoU threshold of 0.5, overlapping bounding boxes signify multiple models flagging the same defect and are counted as "V" number of votes. For instance, if 2 bounding boxes overlap on a defect area, the defect is given 2 votes. And with 3 bounding boxes, that defect is logged as having 3 votes. The assumption driving this strategy is that agreement between disparate models suggests correct prediction. The authors did not rigorously study model choice for ensemble inclusion, instead opting to include all models in the current work.

TABLE III
PER-CLASS PERFORMANCE OF BEST MODELS ON TRAIN/TEST DATA (mAP@IoU=0.5). BOLD INDICATES BEST PERFORMANCE PER ROW

Dataset	Defect Type	YOLOv8 Variant				
		Extra-Small	Small	Medium	Large	Extra-Large
Train	M	0.805	0.885	**0.892**	0.880	0.870
	A	0.928	0.959	**0.969**	0.958	0.959
	P	0.815	0.858	**0.879**	0.835	0.814
	LE	0.662	0.795	0.779	**0.803**	0.776
	LP	0.719	0.824	**0.830**	0.808	0.816
	B	0.934	**0.989**	0.985	0.979	0.976
Test	M	0.653	0.705	**0.765**	0.740	0.716
	A	0.451	0.436	0.477	**0.478**	0.407
	P	0.780	0.785	**0.832**	0.780	0.801
	LE	0.751	0.827	**0.883**	0.817	0.793
	LP	0.835	0.770	**0.849**	0.825	0.846
	B	0.950	0.856	0.901	**1.000**	1.000

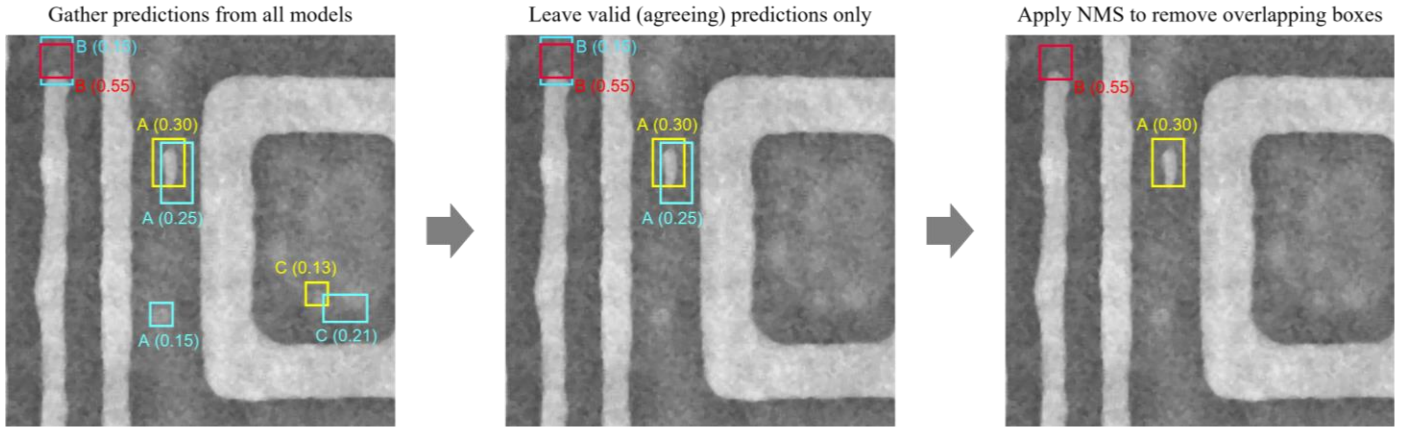

Fig. 5. Ensemble voting strategy for three different model predictions (red, yellow, blue), with a confidence score in parenthesis.

The results of ensemble voting are shown in Table IV. Both the True Positive (TP) and False Positive (FP) rates are improved when moving from single model prediction to the voting classifier. In this work, a prediction is considered as TP if predicted bounding box overlaps at least 30% with ground truth bounding box (i.e., IoU $\geqslant 0.3$), regardless of predicted defect type. Balance between TP and FP degrades significantly with single voters (V=1) and both four and five voter (V=4, V=5) thresholds. This highlights the benefit of choosing the correct number of voters. For practical use, the selection of threshold V can be a practical and quantifiably simple method for creating the "ML recipe" that would run in production in a semiconductor fab facility.

IV. Conclusion

Accurate and timely detection of defects in SEM images is important for yield management. Detection must involve 1) locating the defect, as it provides information about which part of design layout the defect affects, and 2) classifying defect type, as it provides clues to process issues. Results of this paper confirm that when combined into an ensemble voting strategy, such detection can be fully automated with high accuracy based on machine learning models.

TABLE IV
PERFORMANCE FOR SINGLE AND ENSEMBLE MODELS

Single Model					
	Extra Small	Small	Medium	Large	Extra Large
TP Rate (%)	78.8	82.3	85.8	81.4	82.5
FP Rate (%)	15.3	15.4	15.9	14.4	15.9
Ensemble with (V # of votes) Threshold					
	V=1	V=2	V=3	V=4	V=5
TP Rate (%)	92.9	88.8	84.5	77.5	68.4
FP Rate (%)	28.4	15.1	11.6	7.7	5.2

Application of this work can involve seamless integration of the model into process failure identification to increase the accuracy of root cause analysis and faster problem solving by cross-probing between device nodes to view layout, schematic, netlists or by design rules. In future work, prediction of device electrical degradation by detected defects by seamless integration with Calibre® tools will also be explored.

Refences

[1] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A., "You Only Look Once: Unified, Real-Time Object Detection" *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 2016, pp. 779-788 doi: 10.1109/CVPR.2016.91.

[2] Wang, C-Y., Bochkovskiy, A., Liao, H-Y M., "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* 2023, doi: 10.1109/CVPR52729.2023.00721.

[3] Dehaerne, E., Dey, B., Halder, S., and De Gendt, S., "Optimizing yolov7 for semiconductor defect detection," *SPIE: Metrology, Inspection, and Process Control XXXVII*, 2023, pp. 635–642, doi: 10.1117/12.2657564

[4] Yan, S., Ding, S., Wang, S., Luo, C., Li, L., Ai, J., Shen, Q., Xia, Q., Li, Z., Cheng, Q., Li, S., Dai, H. and Hu, X., "Based on deep learning CD-SEM Image Defect Detection System" *China Semiconductor Technology International Conference*, 2022, doi: 10.1109/CSTIC55103.2022.9856857

[5] Dey, B., Dehaerne, E. and Halder, S., "Towards improving challenging stochastic defect detection in SEM images based on improved YOLOv5," *Photomask Technology*, 2022, doi: 10.1117/12.2645402.

[6] Dehaerne, Enrique & Dey, Bappaditya & Esfandiar, Hossein & Verstraete, Lander & Suh, Hyo & Halder, Sandip & De Gendt, Stefan. "YOLOv8 for Defect Inspection of Hexagonal Directed Self-Assembly Patterns: A Data-Centric Approach.", 2023, doi: 10.1117/12.2675573.

[7] J. Kim, Y. Nam, M. Kang, K. Kim, J. Hong, S. Lee, et al., "Adversarial defect detection in semiconductor manufacturing process", *IEEE Trans. Semicond. Manuf.*, 2021, vol. 34, no. 3, pp. 365-371.

[8] Isola, P., Zhu, J-Y., Zhou, T., Efros, A., "Image-to-image translation with conditional adversarial networks". *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017 pp. 5967–5976. doi: 10.1109/CVPR.2017.632