論文 / 著書情報
Article / Book Information

Citation	Hidekazu Takahashi, Hiroki Ogura, Shimpei Sato, Atsushi Takahashi, Chikaaki Kodama "A feature selection method for weak classifier based hotspot detection",Proc. SPIE 11328, Design-Process-Technology Co-optimization for Manufacturability XIV, 113281E,,(2020, 3),doi: https://doi.org/10.1117/12.2559358
Copyright notice	Copyright 2020 Society of Photo-Optical Instrumentation Engineers (SPIE). One print or electronic copy may be made for personal use only. Systematic reproduction and distribution, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper are prohibited.

A feature selection method for weak classifier based hotspot detection

Takahashi, Hidekazu, Ogura, Hiroki, Sato, Shimpei, Takahashi, Atsushi, Kodama, Chikaaki

Hidekazu Takahashi, Hiroki Ogura, Shimpei Sato, Atsushi Takahashi, Chikaaki Kodama, "A feature selection method for weak classifier based hotspot detection," Proc. SPIE 11328, Design-Process-Technology Co-optimization for Manufacturability XIV, 113281E (23 March 2020); doi: 10.1117/12.2559358

Event: SPIE Advanced Lithography, 2020, San Jose, California, United States

A Feature Selection Method for Weak Classifier based Hotspot Detection

Hidekazu Takahashi[a], Hiroki Ogura[a], Shimpei Sato[a], Atsushi Takahashi[a], and Chikaaki Kodama[b]

[a]Tokyo Institute of Technology, Tokyo, Japan
[b]KIOXIA Corporation, Yokohama, Japan

ABSTRACT

As VLSI device feature sizes are getting smaller and smaller, lithography hotspot detection and elimination have become more important to avoid yield loss. Although various machine learning based methods have been proposed, it is not easy to find appropriate parameters to achieve high accuracy. This paper proposes a feature selection method by using the probability distributions of layout features. Our method enables automatic feature optimization and classifier construction. It can be adaptive to different layout patterns with various features. In order to evaluate hotspot detection methods in the situation close to actual problem, dataset based on ICCAD2019 dataset is used for evaluation. Experimental results show the effectiveness of our method and limitations.

Keywords: Lithography, Hotspot detection, Machine Learning

1. INTRODUCTION

In order to form a desired circuit pattern on a wafer, various design for manufacturability (DFM) techniques have been developed. Exposure lights with shorter wavelengths have been developed to improve the fidelity of pattern on a wafer. Although extreme ultra-violet (EUV) whose wavelength is 13.5 nm is being introduced as exposure light, it is still not mainstream in mass production because of huge manufacturing costs, and DFM techniques still have much attention. With the advances of technology nodes, dense pattern with higher yields becomes impossible. Even if layout patterns pass design rule checking (DRC), the patterns may contain local regions that cause yield loss, called hotspots. Lithography hotspot detection and elimination have become more important to improve yields.

Although lithography simulation can be used in hotspot detection, it is very time consuming if the entire chip area is evaluated by accurate lithography simulation. Recently, several hotspot detection methods without lithography simulation have been proposed to reduce the total cost to design a dense chip without hotspots. In these methods, a local chip area, called layout clip, is defined to specify a hotspot, and a huge number of layout clips are defined to cover the entire chip area. Then, a two-class classifier judges whether layout clips are hotspot or not. The total cost will be reduced much if the two-class classifier accurately classifies the clips.

Machine learning can be utilized to construct a two-class classifier. An encoding of a layout pattern that is used to represent the features of the pattern is a key point to improve detection accuracy. Density based layout feature (DBLF)[1,2] is defined by encoding local densities of partitioned sub-regions. Wire distance based layout feature[3] is defined by encoding wire distance in a layout. Histogram of oriented light propagation (HOLP)[4] and concentric circle area sampling (CCAS)[5,6] are expected to capture the impact of diffracted light propagation and interference. Although various layout features are proposed, it is not easy to select appropriate layout features. Deep learning approach is also developed.[7–11] However, it is not easy to select good hyper-parameters as well.

Further author information: (Send correspondence to Hidekazu Takahashi, Atsushi Takahashi)
Hidekazu Takahashi: E-mail: h_takahashi@eda.ict.e.titech.ac.jp
Atsushi Takahashi: E-mail: atsushi@ict.e.titech.ac.jp

Design-Process-Technology Co-optimization for Manufacturability XIV, edited by Chi-Min Yuan, Proc. of SPIE Vol. 11328, 113281E · © 2020 SPIE
CCC code: 0277-786X/20/$21 · doi: 10.1117/12.2559358

In this paper, we propose a CCAS based layout feature selection and optimization method. In our method, the radiuses of circles are selected according to the probability distributions of layout features. Hotspot detection methods based on CCAS are expected to achieve good accuracy as reported in literature. However, in conventional methods, the radiuses of circles to represent layout feature were set manually in advance. It is not necessarily fit to different lithographic conditions. Our proposed method selects appropriate circles that are effective for classification in a specific lithographic condition. In addition, Adaboost technique is utilized in our method to optimize layout features during feature selection. In order to evaluate hotspot detection methods in severe conditions, dataset based on ICCAD2019 dataset is used in experiments.

The rest of the paper is organized as follows. In Section 2, we formulate hotspot detection problem in LSI design and manufacturing. In Section 3, we describe our proposed feature selection method and classifier construction. In Section 4, hotspot detection methods are evaluated by using benchmark datasets, and conclusion is given in Section 5.

2. PRELIMINARIES

2.1 Hotspot Detection

Process variations are no longer negligible in LSI manufacturing. All the process corners and cases cannot be covered by design rules anymore in advanced technology nodes. LSI manufacturing by using perfect design rules that achieves dense patterns with no defects became impossible. Therefore, even if a layout pattern passes design rule checking (DRC), it may contain hotspots where defects occur with some probability inside the process window. To improve yields, it is important to refine the layout pattern as much as possible during design stage by detecting and eliminating hotspots.

2.2 Problem Formulation

In this paper, we focus on hotspot detection problem. The problem is defined as a two-class classification of layout clips. Layout clips are classified into either hotspot or non-hotspot. Layout clip extraction from the layout and hotspot elimination are out of scope in this paper.

A layout clip is a square region whose height and width are $2r_{\max}$ nm and consists of 1 nm^2 square pixels. We assume that a set of layout clips each of which is labeled either hotspot or non-hotspot is given as training data. Let $L = \{x_m \mid m = 1, 2, \ldots, M\}$ be the set of layout clips. Each layout clip is defined as either hotspot or non-hotspot which is represented by label l. Let $l : L \to \{+1, -1\}$ be the label of a layout clip where $l(x_m) = +1$ if x_m is hotspot, $l(x_m) = -1$ otherwise. During our proposed method, each layout clip is assigned a positive weight which is updated iteratively. Let $w_t : L \to \mathcal{R}^+$ be the positive weights of layout clips at round t in the procedure.

In order to form an effective classifier, appropriate feature vectors are needed to be selected. In the proposed methods, feature vectors are formed by concentric circle area sampling (CCAS). The center of a concentric circle is defined on the center of the clip. The radius of concentric circles are defined from 1 nm to $r_{\max}$ nm by 1 nm pitch, and the circle whose radius is i is referred by C_i. Let $C = \{C_i \mid i = 1, 2, \ldots, r_{\max}\}$ be the set of concentric circles defined on layout clip.

The feature vector of a layout clip by circle C_i is formed from p points on C_i at even intervals. At each point, either 0 or 1 is extracted from the layout clip where 0 represents that the point is OFF (black) and 1 that the point is ON (white). Let $z_i^j(x_m)$ be the extracted bit from layout clip x_m at the j-th sampling point on C_i. The feature vector $z_i(x_m)$ of x_m defined by C_i is formed by concatenating extracted bits, that is, $z_i(x_m) = (z_i^1(x_m), z_i^2(x_m), \ldots, z_i^p(x_m))$. The feature vector $z_i(x_m)$ is referred by feature value $f_i(x_m)$ defined as follows:

$$f_i(x_m) = \sum_{j=0}^{p-1} z_i^j(x_m) \times 2^j.$$

Let $V = \{0, 1, \ldots, 2^p - 1\}$ be the range of feature value.

In our method, eight points are selected on a circle. The coordinates of sampling points of C_i are $(\lceil i\alpha \rceil, -\lceil i\alpha \rceil), (0, -i), (-\lceil i\alpha \rceil, -\lceil i\alpha \rceil), (-i, 0), (-\lceil i\alpha \rceil, \lceil i\alpha \rceil), (0, i), (\lceil i\alpha \rceil, \lceil i\alpha \rceil), (i, 0)$ where the clip center is the

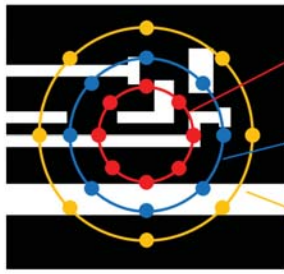


Figure 1: Concentric Circle Area Sampling

origin and $\alpha = \sqrt{2}/2$. For example, the circles in Figure 1 have 8 sampling points, respectively. The corresponding feature values are 17, 242 and 176 in decimal, respectively.

In this paper, hotspot detection methods are evaluated by using following two indexes.

- Recall = #Hotspot-correctly-predicted / #Hotspot
- False Positive Rate (FPR) = #Non-Hotspot-falsely-predicted / #Non-Hotspot

Recall rate is the ratio of hotspots predicted correctly and real hotspots. Lower recall will reduce yield due to missed hotspots. So, recall rate is the most important index. FPR is the ratio of non-hotspots predicted falsely and real non-hotspots. Higher FPR forces unnecessary detailed checking and causes the increase of design cost. The objective in hotspot detection is to achieve high recall rate with low FPR.

3. HOTSPOT DETECTION BY LAYOUT FEATURE SELECTION

In hotspot detection, it is important to use appropriate layout features. Layout hotspots are formed due to the interferences among diffracted lights from photomask. Layout features should have an ability to capture the effect of light propagation and interferences. Concentric circle area sampling (CCAS) is one of the strong candidates to form feature vectors. Although CCAS is originally developed for machine learning based OPC,[5] it is also effective in hotspot detection[6] and in building a regression model to predict light intensity on a wafer.[12] In this section, we discuss feature extraction by using CCAS and a novel feature selection method.

3.1 Feature Vector Evaluation and Selection

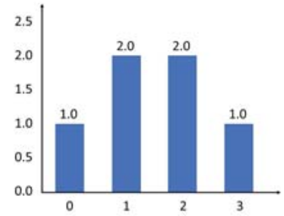
Layout hotspots are caused by an optical proximity effect in the lithography process. CCAS is appropriate layout features because diffracted light propagates concentrically. In the proposed method, several effective concentric circles are selected by using Adaboost algorithm, and a weak classifier is formed for each selected circle at each round. During weak classifiers are defined one by one, clip weights are adjusted to focus on misclassified layout clips in previous classifiers.

Let $p^+_{i,t}(k)$ and $p^-_{i,t}(k)$ be the weighted occurrence probabilities of feature value k of feature vector z_i at round t in hotspot clips and non-hotspot clips in L, respectively. Then, the weighted probability distributions of feature values of feature vector z_i at round t in hotspot clips and non-hotspot clips are defined by $D^+_{i,t} = \{p^+_{i,t}(k) \mid k \in V\}$ and $D^-_{i,t} = \{p^-_{i,t}(k) \mid k \in V\}$, respectively. $D^+_{i,t}$ and $D^-_{i,t}$ are obtained by Algorithm 1.

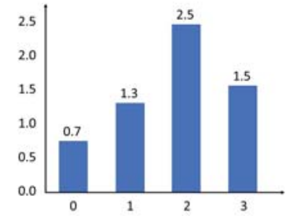
In Figure 2, an illustrative example for the weighted probability distributions of six layout clips where the number of sampling points is 2 is given. In Figure 2a, the feature vector and the feature value of each clip is given

clip	z_i	f_i	w_1	w_t
x_1	(0,0)	0	1.0	0.7
x_2	(1,0)	2	1.0	1.2
x_3	(1,0)	2	1.0	1.3
x_4	(0,1)	1	1.0	0.8
x_5	(1,1)	3	1.0	1.5
x_6	(0,1)	1	1.0	0.5

(a) Clip Information



(b) Probability Distribution (w_1)



(c) Probability Distribution (w_t)

Figure 2: Weighted Probability Distribution

in z_i and f_i column, respectively. Weights given are not normalized for simplicity. The weighted probability distributions when clip weights are given by w_1 and w_t are shown in Figure 2b and Figure 2c, respectively.

If weighted probability distributions $D^+_{i,t}$ and $D^-_{i,t}$ are similar to each other, the ability of feature vector z_i to distinguish hotspot and non-hotspot at round t is expected to be low. In our proposed method, Bhattacharyya coefficient $D^B_{i,t}$ between $D^+_{i,t}$ and $D^-_{i,t}$ defined below is used to evaluate the ability of feature vector z_i at round t:

$$D^B_{i,t} = \sum_{k \in V} \sqrt{p^+_{i,t}(k) \cdot p^-_{i,t}(k)}.$$

The range of $D^B_{i,t}$ is from 0 to 1. The larger $D^B_{i,t}$ is, the more similar the shapes of $D^+_{i,t}$ and $D^-_{i,t}$ are. $D^B_{i,t}$ is 1 if and only if $D^+_{i,t}$ and $D^-_{i,t}$ are identical. The ability of a feature vector is expected to be large if the corresponding $D^B_{i,t}$ is small. Let $b(t)$ be the circle index i at which Bhattacharyya coefficient between distributions $D^+_{i,t}$ and $D^-_{i,t}$ is minimum. That is,

$$b(t) = \arg\min_i D^B_{i,t}.$$

Feature vector $z_{b(t)}$ is regarded as most effective at round t in our definition, and is selected as the input for the weak classifier formed in round t.

3.2 Classifier Construction

In our proposed method, a classifier in which weak classifiers are followed by a strong classifier is formed by Algorithm 2. Let T be the number of weak classifiers which is a user defined parameter.

In our method, weak classifiers are formed one by one. A weak classifier C^w_t is formed at round t. As the input of C^w_t, a feature vector whose Bhattacharyya coefficient is minimum, $z_{b(t)}$, is selected. The output of weak classifier C^w_t is defined according to the weighted probability distributions $D^+_{b(t),t}$ and $D^-_{b(t),t}$. The output of C^w_t for layout clip x is defined as follows:

$$s_t(x) = \ln \frac{p^+_{b(t),t}(k) + \epsilon}{p^-_{b(t),t}(k) + \epsilon}$$

where ϵ is a small constant and $k = f_{b(t)}(x)$. Note that $s_t(x)$ is positive if $p^+_{b(t),t}(k)$ is larger than $p^-_{b(t),t}(k)$, and negative otherwise. Also, if the ratio of difference of them is large, then absolute value is large.

The output of our proposed classifier is defined as the sum of outputs of weak classifiers.

$$s(x) = \sum_{t=1}^{T} s_t(x).$$

The classifier regards the clip as hotspot if the output value exceeds the threshold, and as non-hotspot otherwise.

Each clip weight is initially set to 1, and is adjusted according to the classification results. The weight is reduced if the clip is correctly classified in the previous weak classifier, increased otherwise. Specifically, the following weights are used: $w_1(x_m) = 1$, and $w_{t+1}(x_m) = w_t(x_m) \exp(-l(x_m)s_t(x_m))$ for all layout clip $x_m \in L$.

Algorithm 1 $D^+_{i,t}, D^-_{i,t}$ generation

procedure GENERATE PROBABILITY DISTRIBUTIONS
    for $m = 1$ to M **do**
        if $l(x_m) = +1$ **then**
            $p^+_{i,t}(f_i(x_m)) += w_t(x_m)$
        else if $l(x_m) = -1$ **then**
            $p^-_{i,t}(f_i(x_m)) += w_t(x_m)$
        end if
    end for
    Normalize $D^+_{i,t}$ and $D^-_{i,t}$
end procedure

Algorithm 2 Classifier Construction

Initialize the weights $w_t(x_m)$
procedure FEATURE SELECTION
    for $t = 1$ to T **do**
        for $i = 1$ to r_{max} **do**
            GENERATE PROBABILITY DISTRIBUTIONS $D^+_{i,t}, D^-_{i,t}$
        end for
        Find b_t which is i where $D^B_{i,t}$ is minimum
        Form a weak classifier by using $D^+_{b_t,t}$ and $D^-_{b_t}$
        Update all weights of training clips
    end for
end procedure

4. EVALUATION OF HOTSPOT DETECTION METHODS

In this section, we discuss on the evaluation of hotspot detection methods.

4.1 Dataset

In order to evaluate hotspot detection methods, layout clips with labeled either hotspot or non-hotspot are required. These layout clips are also used to train a classifier in machine learning based methods.

ICCAD2012 dataset has been often used in the evaluation for hotspot detection. The statistics of ICCAD2012 dataset are shown in Table 1. High recall rate and low false positive rate (FPR) were reported for several methods by using ICCAD2012 dataset. However, this dataset was originally developed for pattern matching contest. It contains many identical layout clips and contains few severe clips for hotspot detection. In other words, it can be classified easily, and the performance of methods reported tends to be higher than the actual performance.

While, recently, ICCAD2019 dataset was developed to evaluate hotspot detection in more severe situations.[13] The statistics of ICCAD2019 dataset are shown in Table 2. In ICCAD2019 dataset, the training-set and test1-set consist of some of layout clips in ICCAD2012 dataset, while, the test2-set contains layout clips which are hard to classify. In practice, even a minor nanometers variation in pattern affects hotspot classification. The test2-set contains such layout clips which are near the border between hotspot and non-hotspot.

We prepared benchmark dataset, called SPIE2020 dataset, by rearranging ICCAD2019 dataset. The statistics of SPIE2020 dataset are shown in Table 3. Case1 is generated from the training-set and test1-set in ICCAD2019 dataset by excluding identical layout clips. The training-set and test1-set in ICCAD2019 dataset contain identical clips. Case1 is prepared in order to evaluate the potential of machine learning based methods for unknown layout clips. Case2 and Case3 are part1 and part2 of test2-set in ICCAD2019 dataset, respectively.

Table 1: ICCAD2012 dataset

	Case1	Case2	Case3	Case4	Case5	all_28nm
Techonology	32nm	28nm	28nm	28nm	28nm	28nm
#Training HS	99	174	909	95	26	1204
#Training NHS	340	5285	4643	4452	2716	17096
#Testing HS	226	498	1808	177	41	2524
#Testing NHS	319	4146	3541	3386	2111	13184

Table 2: ICCAD2019 dataset

	Train	Test1	Test2
Techonology	28nm	28nm	28nm
#HS	467	1001	64310
#NHS	17758	14621	65523

Table 3: SPIE2020 dataset

	Case1	Case2	Case3
Techonology	28nm	28nm	28nm
#HS	1008	3860	8469
#NHS	27523	4449	7825

4.2 Evaluation

Our proposed detector is implemented by C++ programming languages, and tested on a machine with four core 4.2 GHz CPUs and 32GB memory.

Experimental results using ICCAD2012 dataset are shown in Table 4. In Table 4, **"M-CPU(s)"**, **"CPU(s)"**, **"#FPs"** and **"Recall(%)"** are the runtime for classifier construction, the runtime for prediction, the number of false positives, and recall rete, respectively. SPIE'15 and SMACD'18 are the method based on the density-encoding[2] and the method based on convolutional neural network (CNN),[7] respectively. The results show that high recall rate and low false positive rate (FPR) are achieved, and that our method achieves comparable performance with other methods.

In Table 5, experimental results using SPIE2020 dataset are shown. A method based on the density-encoding and a method based on convolutional neural network (CNN) are implemented. In the density-encoding based method, local densities are defined on 900 nm^2 regions. In the neural network based method, a network consisting of 4 convolution-pooling layers and 2 fully-connected layers is used. The results are obtained by cross-validation. In each case, layout clips are divided into three subsets, and the average on three tests by using one subset for prediction and the others for training is shown. For Case1, higher Recalls and lower FPRs are observed, and ours achieves the best performance where Recall is 96.4% and FPR is 3.2%. While, for Case2 and Case3, higher FPRs are observed for all methods. This may mean that current layout features used are not good enough to capture the lithographic phenomena effectively.

5. CONCLUSION

As VLSI device feature sizes are getting smaller and smaller, lithography hotspot detection and elimination have become more important to avoid yield loss. This paper proposed a feature selection method by using the probability distributions of layout features. Although our proposed method as well as conventional methods work well for ICCAD2012 dataset, the performance is not good enough for SPIE2020 dataset which includes many layout clips which are near the border between hotspot and non-hotspot. This may mean that current layout features such as density based and CCAS are not good enough to capture the lithographic phenomena effectively. However, our proposed feature selection method and classifier construction could be applied to other types of layout features. We have to analyze the lithographic phenomena to find more appropriate layout features. Hotspot detection is still challenging and significant effort is required to achieve enough yield with less design and manufacturing costs.

Table 4: Experimental Results using ICCAD2012 dataset

	SPIE'15[2]			SMACAD'18[11]			Ours			
	CPU(s)	#FPs	Recall(%)	CPU(s)	#FPs	Recall(%)	M-CPU(s)	CPU(s)	#FPs	Recall(%)
Case1	7.0	0	100.0	N/A	0	100.0	0.02	3.9	0	98.7
Case2	351.0	0	98.6	N/A	7	99.8	0.63	84.8	2	98.2
Case3	297.0	0	97.2	N/A	3	99.8	0.64	106.5	0	98.9
Case4	170.0	1	87.0	N/A	2	99.8	0.52	47.7	0	95.5
Case5	69.0	0	92.9	N/A	1	95.1	0.20	28.0	0	95.1
Average	178.8	0.2	95.1	N/A	2.6	98.9	0.40	54.2	0.4	97.3

Table 5: Experimental Results using SPIE2020 dataset

	Density		CNN		Ours	
	Recall(%)	FPR(%)	Recall(%)	FPR(%)	Recall(%)	FPR(%)
Case1	64.3	1.9	84.1	2.5	96.4	3.2
Case2	83.2	19.4	85.1	19.2	84.8	27.8
Case3	86.9	21.6	90.0	13.7	83.7	32.7

REFERENCES

[1] Wuu, J., Pikus, F. G., Torres, A., and Marek-Sadowska, M., "Rapid layout pattern classification," in [*Asia and South Pacific Design Automation Conference (ASP-DAC)*], 781 – 786 (2011).

[2] Matsunawa, T., Gao, J.-R., Yu, B., and Pan, D., "A new lithography hotspot detection framework based on AdaBoost classifier and simplified feature extraction," in [*Proc. SPIE Advanced Lithography*], **9427**, 201 – 211 (2015).

[3] Kataoka, G., Inagi, M., Nagayama, S., and Wakabayashi, S., "Novel feature vectors considering distances between wires for lithography hotspot detection," in [*Euromicro Conference on Digital System Design (DSD)*], 85 – 90 (2018).

[4] Tomioka, Y., Matsunawa, T., Kodama, C., and Nojima, S., "Lithography hotspot detection by two-stage cascade classifier using histogram of oriented light propagation," in [*Asia and South Pacific Design Automation Conference (ASP-DAC)*], 81 – 86 (2017).

[5] Matsunawa, T., Yu, B., and Pan, D., "Optical proximity correction with hierarchical Bayes model," in [*Proc. SPIE Advanced Lithography*], **9426**, 238 – 247 (2015).

[6] Zhang, H., Yu, B., and Young, E. F. Y., "Enabling online learning in lithography hotspot detection with information-theoretic feature optimization," in [*IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*], 1 – 8 (2016).

[7] Shin, M. and Lee, J.-H., "Accurate lithography hotspot detection using deep convolutional neural networks," *Journal of Micro/Nanolithography, MEMS, and MOEMS* **15**(4), 1 – 13 (2016).

[8] Yang, H., Luo, L., Su, J., Lin, C., and Yu, B., "Imbalance aware lithography hotspot detection: a deep learning approach," *Journal of Micro/Nanolithography, MEMS, and MOEMS* **16**(3), 1 – 13 (2017).

[9] Yang, H., Lin, Y., Yu, B., and Young, E. F. Y., "Lithography hotspot detection: From shallow to deep learning," in [*IEEE International System-on-Chip Conference (SOCC)*], 233 – 238 (2017).

[10] Yang, H., Su, J., Zou, Y., Yu, B., and Young, E. F. Y., "Layout hotspot detection with feature tensor generation and deep biased learning," in [*ACM/IEEE Design Automation Conference (DAC)*], 1 – 6 (2017).

[11] Borisov, V. and Scheible, J., "Lithography hotspots detection using deep learning," in [*International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*], 145 – 148 (2018).

[12] Kimura, T., Matsunawa, T., Nojima, S., and Pan, D., "Hybrid hotspot detection using regression model and lithography simulation," in [*Proc. SPIE Advanced Lithography*], **9781**, 107 – 116 (2016).

[13] Reddy, G. R., Madkour, K., and Makris, Y., "Machine learning-based hotspot detection: Fallacies, pitfalls and marching orders," in [*IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*], 1 – 8 (2019).