Enhancements of Model and Method in Lithography Hotspot Identification

Xuanyu Huang*
Department of Mechanical Engineering
Center for Nano and Micro Mechanics
Tsinghua University
Beijing, China
huangxua20@mails.tsinghua.edu.cn

Rui Zhang*
HiSilicon Technologies Co., Ltd.
Shenzhen, China
zhangrui.semi@hisilicon.com

Yu Huang
HiSilicon Technologies Co., Ltd.
Shenzhen, China
huangyu61@hisilicon.com

Peiyao Wang
HiSilicon Technologies Co., Ltd.
Shenzhen, China
wangpeiyao@hisilicon.com

Mei Li
HiSilicon Technologies Co., Ltd.
Shenzhen, China
limei.limei@hisilicon.com

***Abstract*—The manufacturing of integrated circuits (ICs) has been continuously improved through the advancement of fabrication technology nodes. However the lithography hotspots (HSs) caused by optical diffraction problems seriously affect the yield and reliability of ICs. Although lithography simulation can accurately capture the HSs through physically simulating the lithography process, it requires a lot of computing resources, which usually takes > 100 CPU $\cdot h/mm^2$ [1]. Due to the image recognition nature, the state-of-the-art HS identification algorithms based on deep learning have obvious advantages in reducing run time comparing to the traditional algorithms. However, its accuracy still needs to be enhanced since there are many false alarms of non-hotspots (NHSs) and escapes of the real HSs, which makes it difficult to be a signoff technique. In this paper, we propose two enhancements in HS identification. First, a hybrid deep learning model is proposed in lithography HS identification, which includes a CNN model to combine physical features. Second, an ensemble learning method is proposed based on multiple sub-models. The proposed enhanced model and method can achieve high HS identification accuracy on the benchmarks 1-4 of the ICCAD 2012 dataset with recall$> 98.8\%$. In addition, it can achieve even 100% recall on the benchmark 1 and benchmark 3 while maintaining the precision at a high level with 53.6% and 87.1%, respectively. Moreover, for the first time it can achieve not only 100% recall on benchmark 5, but also high precision of 61.8%, which is much higher than any published deep learning methods for HSs identification, as far as we know. The proposed model and methodology can be applied in industrial IC designs due to its effectiveness and efficiency.**

***Index Terms*—lithography hotspot, ensemble learning, deep learning, convolutional neural network**

I. Introduction

The critical dimensions of devices in VLSI have been continuously scaling down through the micro-nano fabrication technology advancement. Consequently the feature sizes of new technology nodes have been continuously shrinking in pursuit of better performance. Lithography is the most critical patterning process in the VLSI fabrication flow. However, the optical diffraction effect becomes more and more significant due to the shrinking feature size. It causes the actual light intensity non-uniformly distributing across the silicon layout, which may easily lead to fabrication failures (such as metal bridging or segment open problems) at some special layout locations, which are typically called lithography hotspots (HSs). HSs can reduce the yield of the production significantly and weak patterns are threats to the chip reliability. As a matter of fact, many foundries proposed different design rules checks (DRCs), which can eliminate some of the HSs. Nevertheless in the real world of design and fabrication, some HSs are still created even though all DRCs are satisfied during physical design. Therefore, the identification of HSs is a very important task in VLSI physical design.

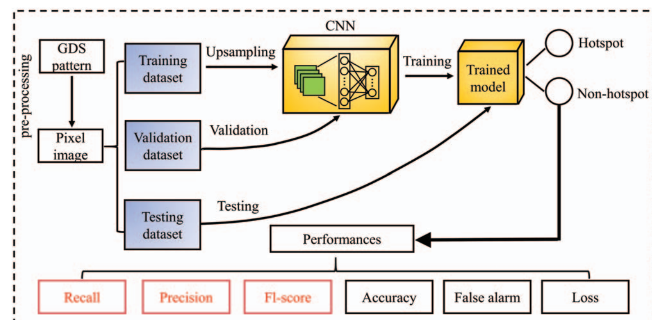

Fig. 1. Overall organization and process of CNN-based lithography hotspot identification algorithm flow.

Rigorous lithography simulation is an accurate method to capture HSs because it can physically simulate the lithography process. However, it requires a lot of computing resources. Model-based lithography verification usually requires > 100 CPU $\cdot h/mm^2$ [1], and an accurately calibrated model preparation may take several days [2], which consumes a lot of CPU resources and incurs cost overhead. Therefore, fast and accurate algorithms for HSs identification at design level become

*These two authors contributed equally to this work.

978-3-9819263-5-4/DATE21/©2021 EDAA

absolutely necessary.

The lithography HSs primarily depend on the geometric information in the layout. Hence prior work introduced some geometric algorithms to analyze the layout geometric characteristics, rather than relying on the rigorous optical physical simulation. These methods depend entirely on the layout engineers' experiences of lithography HSs and non-hotspots (NHSs), such as pattern matching methods [3]–[6], machine learning methods [7]–[12], and fuzzy matching methods [13], [14], etc. Among them, the pattern matching method proposed in [3]–[6] cannot predict HSs beyond the information stored in its HSs database, which leads to the low accuracy of HSs identification. The machine learning methods consist of complex feature engineering, which are difficult to be generalized and applied to different designs. When running the machine learning methods proposed in [4], [8] on the ICCAD 2012 dataset [15], they suffered from many false alarms (predicting NHs as HSs). The fuzzy matching method proposed in [13] had low identification rate and identification performance fluctuation. In addition, it was extremely dependent on the characteristics of the datasets [14].

As we all know, convolutional neural networks (CNN) show very good results in image recognition and classification [16], [17], because of its good generalization and perception capabilities. Therefore, the application of CNN in the identification of HSs [2], [18] has achieved good results, it not only has a lower false alarm rate and higher accuracy, but also due to the simplicity of its forward propagation, its inference time is shorter than other method [2], [18], [19]. The typical reported CNN flow for hotspot identification mainly include three steps. First, the GDS layout information is converted into array form data and processed appropriately. Next, the processed data is applied to the designed CNN network for training process. Finally the trained network is used to identify the test data set, and can obtain the performance of the CNN model. However, there are still some problems for applying such techniques in the realistic applications. First, they cannot detect all HSs on the dataset in ICCAD 2012 [15], such as benchmark 5 [14]. Secondly, the false alarm rates of these methods are high. For example, the false alarm of the reported CNN method is 269 on benchmark 5 (total 19327 NHSs) in ICCAD 2012 dataset [19], which leads to only 12.9% precision. Considering that in the real chips there are more than hundreds of thousand patterns, which is several orders of magnitude higher than the number of patterns in ICCAD 2012 [2], [18], it means that the prior published methods cannot be applied in realistic chip design projects unless the false alarm rate is reduced.

In this paper, two enhancements are proposed. First, we propose a hybrid deep learning model in lithography HS identification, which applies a multi-input CNN model to incorporate physical features. Next, we propose an ensemble learning method using multiple sub-models. The remainder of this paper is organized as follows. Section II will propose a novel CNN model and some initial experimental results are illustrated. Section III will explain how to enhance the proposed model by incorporating physical features and how to use an ensemble method to improve the performance. More experimental results are presented in this section. Section IV will conclude the paper.

II. The proposed CNN model structure and initial experimental results

The proposed model is shown in Fig. 1, which is mainly consisted of three parts: (i) The first part is for the pre-processing, which converts the coordinate information in the GDS layout into the pixel data that CNN can recognize. The data will be divided into three parts: training dataset, validation dataset and test dataset. (ii) The second part is for the training process. The proposed CNN model interprets HS related geometric features from training dataset through representation learning, and the validation dataset is used to determine whether the model has reached the expected accuracy. Note that the validation dataset is not used in the random gradient descent algorithm during the training process. Finally, after the training process, a trained model is ready to classify the input images as HS or NHS. (iii) The third part is the test process, which predicts the testing dataset with the same characteristics as training datasets, and reports the coordinates for HSs. In this work, we focus on the enhancement of CNN model to achieve better HS identification performance.

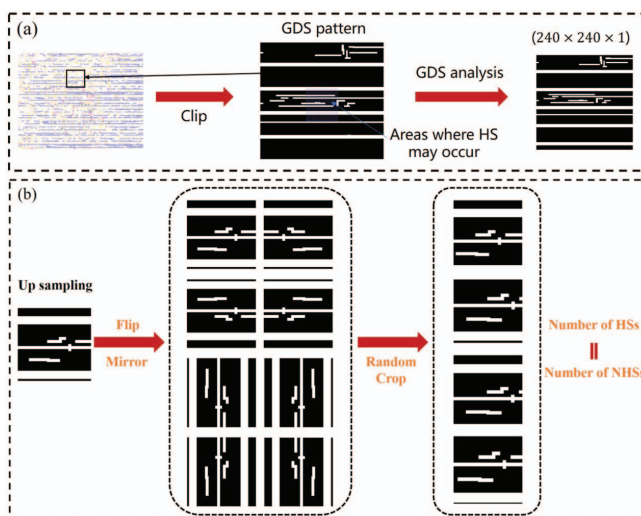

Fig. 2. Data processing (a) Data pre-processing. (b) Data augmentation (Up sampling).

The schematic diagram of the proposed data pre-processing is illustrated in Fig. 2(a). In this step, the dataset provided by the ICCAD2012 competition [15] is used. It includes five benchmarks using 32nm and 28nm fabrication technology nodes. Each benchmark has different amount of HS/NHS data in the train/test dataset. Firstly, it partitions a big GDS layout into smaller patterns. Secondly, it reads the coordinates of the vertices of the polygon, and then converts all the polygons into pixel images of the size of (240, 240, 1). It is worth noting that this dataset has unbalanced characteristics, and as the number of benchmarks increases, the degree of imbalance continues to increase. For example, in the training dataset of benchmark 5

$HS : NHS = 1 : 104$. Such an imbalance causes challenges for the training process of CNN models [20]–[22]. Therefore, we need to use external intervention to assist the model to learn the characteristics of HS from unbalanced data. At the dataset level, a data augmentation algorithm is proposed to use mathematic methods to generate new HS images. Although there are many ways to construct a data augmentation, due to the images used in this context are binary images, many perturbation methods are not applicable because they require continuous pixel values. Meanwhile, the lithography HSs are also very sensitive to the spacing between polygons. Hence, some zooming and panning methods will destroy the features of the original HSs. Due to the reasons mentioned above, a data augmentation flow is proposed with a flipping followed by random cropping to up-sampling, as illustrated in Fig. 2(b). It flips each HS image data to generate up to eight pictures, and then randomly crop each picture as needed to make sure that the middle area, where HS occurs, is not cut out during the cropping process.

Considering the speed of training and prediction process of CNN model, we use Mobilenet network's basic structure [23], which splits the complete convolution layer into depthwise convolution layer and pointwise convolution to greatly speed up the training and the prediction processes. The input of the network is an image data with a size of $(240, 240, 1)$ pixels, and the output is a HS probability p_{HS} in the range of [0,1]. Since the main purpose of this paper is to identify and extract the HSs from the layout patterns, in order to simplify the model, we treat it as a binary classification problem, and the threshold is set to 0.5, that is, when $0.5 \le p_{HS} \le 1$, the network predicts the input image data as HS, otherwise it is predicted as NHS. During the training process, we use a stochastic gradient descent algorithm based on the backpropagation to update the weight W of the network

$$W_{i+1} = W_i - \xi \left(\frac{\partial L_i}{\partial W} \right) \tag{1}$$

where ξ is the learning rate and L_i is the loss value of each epoch calculated by focal-loss function [24].

$$\begin{aligned} L_i^{HS}(p_{HS}) &= -\alpha_{HS}(1-p_{HS})^{\gamma} log(p_{HS}), \\ L_i^{NHS}(p_{HS}) &= -\alpha_{HS}(p_{HS})^{\gamma} log(1-p_{HS}) \end{aligned} \tag{2}$$

where L_i^{HS} and L_i^{NHS} are the loss function of HS and NHS data respectively, α_{HS} is the label weight of HS, which can make the model pay more attention to HS data to reduce the impact of data imbalance on the training process, and $(1 - p_{HS})^{\gamma}$ item used in L_i^{HS} can make model pays more penalty to the data with wrong predictions ($p_{HS} < 0.5$), which γ is the degree of differentiation.

Based on the above-proposed algorithm and method, we calculate the performance by running on each benchmark on the ICCAD dataset as shown in Table I setting $\gamma = 2$ and $\alpha_{HS} = 0.7$. We use three indicators fl-score, recall and precision to judge the prediction performance of the trained model due to the imbalance of the dataset.

$$\begin{aligned} recall &= \frac{N_{THS}}{N_{HS}}, \\ precision &= \frac{N_{THS}}{N_{PHS}}, \\ fl-score &= \frac{2recall \times precision}{recall + precision} \end{aligned} \tag{3}$$

where N_{THS} is the number of correctly predicted HSs, N_{HS} is the total number of HSs, and N_{PHS} is the number predicted by the model as HSs. Therefore, the recall is the proportion of all HSs that are successfully identified, which can reflect the success rate of the HS identification by the proposed model. The precision is the proportion of correctly identified patterns by using the model, which can reflect the recognition degree of the model. The lower precision indicates higher false alarm rate, and the fl-score is a comprehensive index combining the recall and the precision. The best recall, best precision and best fl-score are the models with the highest corresponding indexes during the training process. It can be seen from the table that we can achieve extremely high recall in benchmarks 1-5 (even achieving recall=1, i.e., all HSs are detected), and can also guarantee extremely high precision. For example, the benchmarks 1-4 can achieve more than 98% precision when recall $> 95\%$, that is, fl-score$> 96\%$. In comparison, the best recall model of the state-of-the-art on benchmarks 1-4 had much lower precision of $< 47\%$ [26]. For benchmark 5 with the highest imbalance and the closest scenario to the realistic situation, the proposed model can still detect all HSs, with the precision to be optimized to 44.2%. In comparison, the highest recall could be achieved by state-of-the-art on benchmark 5 was only 97.6% [19], which means that not all the HSs could be detected. Furthermore, the corresponding precision was also as low as 12.9%, which means there were too many false alarms such as it could correctly identify 1 HS together with about 7 false alarms.

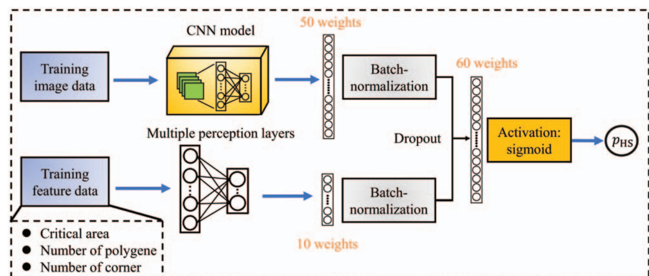

Fig. 3. The diagram of multi-input CNN model algorithm flow.

III. The Proposed Model and Method Enhancements

Although the above-proposed algorithm has achieved good performance in benchmarks 1-4, the precision of benchmark 5 still needs to be improved, because it will produce an unacceptable number of false alarms if it is applied to a realistic chip design with a larger data number. In this session, we propose two enhancements and their effects in Sections III-A and III-B.

TABLE I
COMPARISON OF THE TEST PERFORMANCE OF EACH BENCHMARK ON THE ICCAD DATASET WITH THE EXISTING WORKS WHEN $\gamma = 2$ AND $\alpha_{HS} = 0.7$:

	Model	**Our works**			**Existing works**			
		recall	precision	fl-score	Ref.	recall	precision	fl-score
Benchmark 1	Best recall	**1.000**	0.536	0.698	[25]	**1.000**	0.223	0.365
	Best precision	0.696	**1.000**	0.821	[26]	0.996	**0.605**	0.753
	Best fl-score	0.996	0.996	**0.996**	[26]	0.996	0.605	**0.753**
Benchmark 2	Best recall	**0.988**	0.974	0.981	[26]	**0.998**	0.470	0.639
	Best precision	0.958	**0.996**	0.977	[18]	0.987	**0.856**	0.917
	Best fl-score	0.974	0.994	**0.984**	[18]	0.987	0.856	**0.917**
Benchmark 3	Best recall	**1.000**	0.817	0.931	[18]	**0.980**	0.358	0.524
	Best precision	0.998	**0.997**	0.998	[25]	0.975	**0.462**	0.627
	Best fl-score	0.998	0.997	**0.998**	[25]	0.975	0.462	**0.627**
Benchmark 4	Best recall	**0.995**	0.726	0.839	[19]	**0.994**	0.047	0.089
	Best precision	0.937	**0.994**	0.965	[18]	0.945	**0.361**	0.522
	Best fl-score	0.952	0.984	**0.968**	[18]	0.945	0.361	**0.522**
Benchmark 5	Best recall	**1.000**	0.442	0.613	[19]	**0.976**	0.129	0.229
	Best precision	0.952	**0.769**	0.851	[25]	0.951	**0.293**	0.448
	Best fl-score	0.952	0.769	**0.851**	[25]	0.951	0.293	**0.448**

A. Multi-input CNN model incorporating physical feature data

The advantage of the convolutional neural network and deep learning algorithm is that it has a certain depth perception ability for image data, which can learn local features to make accurate predictions. However, CNN is based on purely geometric information for representation learning, rather than considering the physical information related to lithography. Therefore, it is inefficient to capture the physical features of HS through CNN itself. The HSs are physically caused by diffraction in lithography, so it is highly correlated with the critical parameters of the layout. Based on the above observations, we propose a multi-input model, which combines the advantages of CNN and feature extraction to achieve higher representation learning efficiency. As illustrated in Fig. 3, the original single image data input is augmented with the second feature data input branch. The original CNN model still processes the data of the pixel images as the first input branch, where it will calculate 50 weights as outputs. The second input branch with the feature data are fed through multiple perception layers (MPLs), which will calculate another 10 weights as outputs. The feature data is obtained by going through additional scripts during the GDS analysis process to extract the geometric feature of each pattern related to lithography. Taking the ICCAD dataset as an example, we first extract the number of polygons, the number of corners and the critical area in each pattern.These information are inferred based on the characteristics of the lithography process. Finally, these 60 weights are combined through the batch-normalization operation, and the final probability p_{HS} is calculated through the last fully connect layer with sigmoid activation.

We apply the above-mentioned method to benchmark 5 of ICCAD dataset with $\gamma = 2$ and $\alpha_{HS} = 0.7$, the precision, recall and loss in the training process related to epoch are shown in Fig. 4(a). The black curve indicates the training dataset and red curve indicates the validation dataset. It can be seen from this diagram that after 7 training epochs, the recall on the validation dataset is converged to 1, and the precision is gradually increased in the subsequent training epochs. Finally it is converged after 25 epochs. We select the model with the

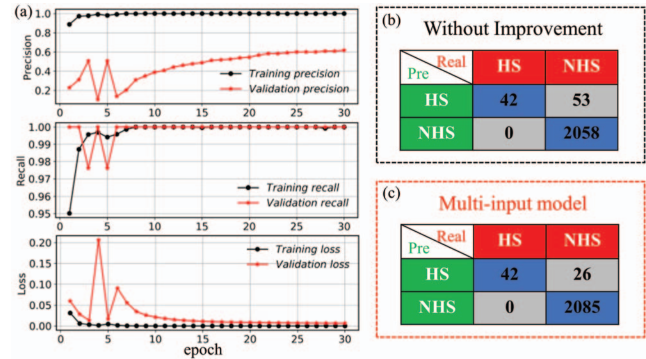

Fig. 4. The training process and performance of multi-input method. (a) The precision (upper), recall (middle) and loss (lower) in each epoch, the black line indicates the training dataset and red line indicate the validation dataset. (b) The summary of prediction results (confusion matrix) of the model without use multi-input method. (d) The confusion matrix of the model uses multi-input method.

highest recall and test its performance on the testing dataset and compare the results with the previous model without physical feature data input. The summaries of prediction results, i.e., the confusion matrices are shown in Fig. 4(b) for before the enhancement vs. Fig. 4(c) for after the enhancement. It can be seen that the performance of the model has been significantly improved after adding the input of physical feature parameters. It indicates that not only the recall can achieve 100%, but also the precision can be improved from 44.21% to 61.76% (due to the number of false alarms is decreased from 53 to 26). This hybrid model can be applied for future realistic chip design.

B. Ensemble learning method

Ensemble learning [27]–[29] is the process by which multiple models, such as classifiers or experts, are strategically generated and combined to solve a particular computational intelligence problem. Ensemble learning is primarily used to improve the (classification, prediction, function approximation, etc.) performance of a model, or reduce the likelihood of an unfortunate selection of a poor one. Other applications of ensemble learn-

TABLE II
COMPARISON OF BEFORE AND AFTER MODEL IMPROVEMENT AND EXISTING WORK ON BENCHMARK 5 WHEN $\gamma = 2$ AND $\alpha_{HS} = 0.7$:

Works	Best recall model for benchmark 5		
	recall	precision	fl-score
Existing works [19]	0.976	0.129	0.229
Our work without improvement	1.000	0.442	0.613
Multi-input method	1.000	0.618	0.764
Ensemble learning method	1.000	0.525	0.689

ing include assigning a confidence to the decision made by the model, selecting optimal (or near optimal) features, data fusion, incremental learning, non-stationary learning and error-correcting. In our application, it could effectively ensemble the advantages and make up for the shortcomings of multiple sub-networks, and could provide the possibility with higher accuracy than a single network.

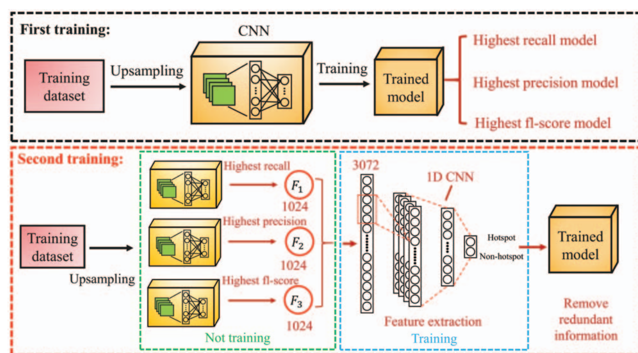

Fig. 5. The diagram of ensemble learning method algorithm flow for HS identification.

For our model in the training process, CNN is equivalent to finding the boundary of HSs and NHSs. Hence the recall and precision of the model calculated during the training process will form a contradiction. If the model is aggressive, it may be able to catch more HSs, but may cause many false alarms such that many NHSs with similar characteristics as HSs, and vice versa. If the model is too conservative, lots of HSs may be missed. Therefore, the ensemble learning method is proposed to combine models with their own characteristics, which could combine the advantages of each model. The algorithm flow is shown in Fig. 5. This method requires two training processes and the first training is the same method as described in Section II. After the first training is done, we could select three trained models which have the highest recall, the highest precision, and the highest fl-score. In the second training, we extract the middle fully connected layer of these models corresponding to 1024 weights respectively, and then combine the weights as the input of the next one-dimensional convolutional neural network (1D CNN) [30], [31]. This 1D CNN is mainly composed of a convolutional layer, a pooling layer and a fully connected layer. The purpose of this 1D CNN is to select optimal weights from the combined models and remove redundant weights [30], [31]. It's worth noting that in the second training process, the weights in the models selected in first training will not be trained. It

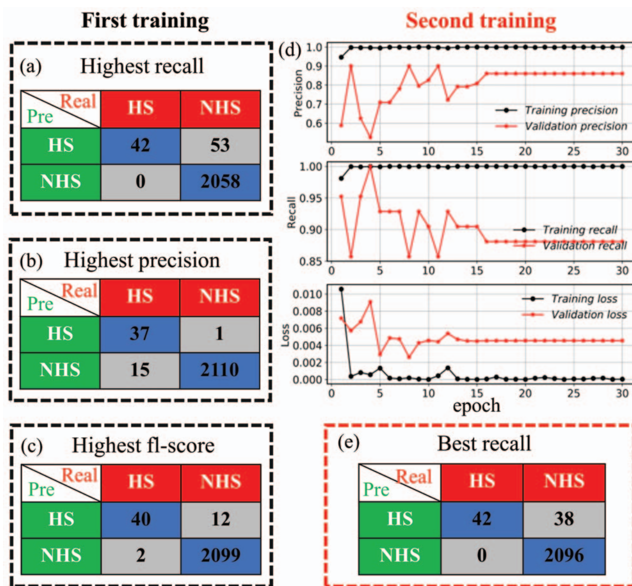

Fig. 6. The performance of ensemble learning method in benchmark 5 when $\gamma = 2$ and $\alpha_{HS} = 0.7$: (a)-(c) The confusion matrix of selected three models with the highest recall, fl-score, and precision in the first training process. (d)The precision (upper), recall (middle) and loss (lower) in each epoch of the second training process, the black line indicates the training dataset and red line indicate the validation dataset. (e) The confusion matrix of the best recall model for second training process.

only trains the following 1D CNN's weights to get a coupled model. We apply the above-proposed method to benchmark 5 of ICCAD dataset with $\gamma = 2$ and $\alpha_{HS} = 0.7$. The first training process is completed based on the flow described in Section II. Three models with highest recall, precision, and fl-score are selected, respectively. Their confusion matrices on testing dataset are shown in Fig. 6 (a)-(c). In the second training process, we train the weights of 1D CNN, where the precision, recall and loss in each epoch are shown in Fig. 6(d). The black curve indicates the training dataset and red curve indicates the validation dataset. It can be seen that 1D CNN converges after around 16 epoch. We finally select the model with the highest recall in the second training process and test it on the testing dataset. The confusion matrix is shown in Fig. 6(e). Comparing to the model without using the ensemble learning method in Fig. 4(b), this model not only can achieve 100% recall, but also can improve the precision from 44.21% to 52.50% (due to the number of false alarms is decreased from 53 to 38). This method can be incorporated in the multi-input model described

in Section III-A to achieve higher performance.

By using benchmark 5, an analytic comparison across the state-of-the-art vs. 3 models proposed in this paper is shown in Table II. It can be seen that the proposed model is very sensitive to HS and can achieve 100% recall comparing to the state-of-the-art. With the enhancements of the multi-input model and ensemble learning method, the precision is further improved and the false alarm rate is greatly reduced, which means that our proposed enhancements can be applied to more realistic design with more patterns.

IV. Conclusions

In this paper, two enhancements were proposed. We first proposed a hybrid deep learning model in lithography HS identification, which incorporates a multi-input CNN model combining the layout images with physical feature data. Next, we proposed an ensemble learning method. The proposed method can achieve high HS identification rate with recall$>98.8\%$ on the first 4 benchmarks of the ICCAD 2012 dataset, and even with recall $= 100\%$ on benchmarks 1 and 3 while maintaining the precision at a high level with 53.6% and 87.1%, respectively. In addition, for the first time we can achieve 100% recall on benchmark 5. In addition, through the multi-input model with physical feature data and an ensemble learning method it can achieve a high precision of 61.8% and 52.50% respectively, which is much higher than the state-of-the-art deep learning method for the HSs identification. These enhancements lays the foundation for the deep learning-based HS identification methods to be applied to big industrial IC designs in future.

Acknowledgment

We would like to express our gratitude to Chen Cheng, Tianzhen Li, Jingxia Liu, Weiping Xiao for their professional comments on this work. Besides, we wish to give a special thank you to Huatao Yu for his constant support and guidance to this work.

References

[1] W. Jen-Yi, F.G. Pikus, A. Torres, M. Marek-Sadowska, Rapid layout pattern classification, 2011.

[2] M. Shin, J.-H. Lee, Accurate lithography hotspot detection using deep convolutional neural networks, Journal of Micro-Nanolithography Mems and Moems, 15 (2016).

[3] A.B. Kahng, C.-H. Park, X. Xu, Fast dual-graph-based hotspot filtering, Ieee Transactions on Computer-Aided Design of Integrated Circuits and Systems, 27 (2008) 1635-1642.

[4] J. Xu, S. Sinha, C.C. Chiang, Ieee, Accurate detection for process-hotspots with vias and incomplete specification, Ieee/Acm International Conference on Computer-Aided Design Digest of Technical Papers, Vols 1 and 22007, pp. 839-846.

[5] H. Yao, S. Sinha, C. Chiang, X. Hong, Y. Cai, Ieee, Efficient process-hotspot detection using range pattern matching, 2006.

[6] Y.-T. Yu, Y.-C. Chan, S. Sinha, I.H.-R. Jiang, C. Chiang, Ieee, Accurate Process-Hotspot Detection Using Critical Design Rule Extraction, 2012 49th Acm/Edac/Ieee Design Automation Conference2012, pp. 1163-1168.

[7] D. Ding, J.A. Torres, D.Z. Pan, High Performance Lithography Hotspot Detection with Successively Refined Pattern Identifications and Machine Learning, Ieee Transactions on Computer-Aided Design of Integrated Circuits and Systems, 30 (2011) 1621-1634.

[8] D. Ding, X. Wu, J. Ghosh, D.Z. Pan, Ieee, Machine Learning based Lithographic Hotspot Detection with Critical-Feature Extraction and Classification, 2009.

[9] T. Matsunawa, J.R. Gao, B. Yu, D.Z. Pan, A New Lithography Hotspot Detection Framework Based on AdaBoost Classifier and Simplified Feature Extraction, in: J.L. Sturtevant, L. Capodieci (Eds.) Design-Process-Technology Co-Optimization for Manufacturability Ix2015.

[10] N. Nagase, K. Suzuki, K. Takahashi, M. Minemura, S. Yamauchi, T. Okada, Study of hotspot detection using neural networks judgment - art. no. 66071B, in: H. Watanabe (Ed.) Photomask and Next-Generation Lithography Mask Technology Xiv, Pts 1 and 22007, pp. B6071-B6071.

[11] B. Yu, J.-R. Gao, D. Ding, X. Zeng, D.Z. Pan, Accurate lithography hotspot detection based on principal component analysis-support vector machine classifier with hierarchical data clustering, Journal of Micro-Nanolithography Mems and Moems, 14 (2015).

[12] Y.-T. Yu, G.-H. Lin, I.H.-R. Jiang, C. Chiang, Machine-Learning-Based Hotspot Detection Using Topological Classification and Critical Feature Extraction, Ieee Transactions on Computer-Aided Design of Integrated Circuits and Systems, 34 (2015) 460-470.

[13] L. Sheng-Yuan, C. Jing-Yi, L. Jin-Cheng, W. Wan-yu, C. Shih-Chieh, A novel fuzzy matching model for lithography hotspot detection, 2013.

[14] W.-Y. Wen, J.-C. Li, S.-Y. Lin, J.-Y. Chen, S.-C. Chang, A Fuzzy-Matching Model With Grid Reduction for Lithography Hotspot Detection, Ieee Transactions on Computer-Aided Design of Integrated Circuits and Systems, 33 (2014) 1671-1680.

[15] J.A. Torres, Ieee, ICCAD-2012 CAD Contest in Fuzzy Pattern Matching for Physical Verification and Benchmark Suite, 2012 Ieee/Acm International Conference on Computer-Aided Design2012, pp. 349-350.

[16] D.C. Ciresan, A. Giusti, L.M. Gambardella, J. Schmidhuber, Mitosis Detection in Breast Cancer Histology Images with Deep Neural Networks, in: I. Sakuma, C. Barillot, N. Navab (Eds.) Medical Image Computing and Computer-Assisted Intervention - Miccai 2013, Pt Ii2013, pp. 411-418.

[17] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Communications of the ACM, 60 (2017) 84-90.

[18] H. Yang, L. Luo, J. Su, C. Lin, B. Yu, Imbalance aware lithography hotspot detection: a deep learning approach, Journal of Micro-Nanolithography Mems and Moems, 16 (2017).

[19] W. Lu, Y. Zhang, Q. Zhang, X. Zhang, Y. Li, Litho-NeuralODE: Improving Hotspot Detection Accuracy with Advanced Data Augmentation and Neural Ordinary Differential Equations, Proceedings of the 2020 on Great Lakes Symposium on VLSI, 2020, pp. 387-392.

[20] Learning from Imbalanced Data Sets. Papers from the AAAI Workshop (Technical Report WS-00-05), 2000.

[21] N.V. Chawla, N. Japkowicz, A.J.A.S.e.n. Kotcz, Special issue on learning from imbalanced data sets, 6 (2004) 1-6.

[22] F. Provost, Machine learning from imbalanced data sets, 2000.

[23] A.G. Howard, Z. Menglong, C. Bo, D. Kalenichenko, W. Weijun, T. Weyand, M. Andreetto, H. Adam, MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications arXiv, arXiv, DOI (2017) 9 pp.-9 pp.

[24] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollar, Ieee, Focal Loss for Dense Object Detection, 2017 Ieee International Conference on Computer Vision2017, pp. 2999-3007.

[25] H. Zhang, B. Yu, E.F.Y. Young, Acm, Enabling Online Learning in Lithography Hotspot Detection with Information-Theoretic Feature Optimization, 2016 Ieee/Acm International Conference on Computer-Aided Design2016.

[26] H. Yang, L. Lu, J. Su, C. Lin, B. Yu, Imbalance Aware Lithography Hotspot Detection: A Deep Learning Approach, in: L. Capodieci, J.P. Cain (Eds.) Design-Process-Technology Co-Optimization for Manufacturability Xi2017.

[27] Y. Freund, R.E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, Journal of Computer and System Sciences, 55 (1997) 119-139.

[28] L. Breiman, Bagging predictors, Machine Learning, 24 (1996) 123-140.

[29] A. Yazdizadeh, Z. Patterson, B. Farooq, Ensemble Convolutional Neural Networks for Mode Inference in Smartphone Travel Survey, Ieee Transactions on Intelligent Transportation Systems, 21 (2020) 2232-2239.

[30] Y. Aytar, C. Vondrick, A. Torralba, SoundNet: Learning Sound Representations from Unlabeled Video, in: D.D. Lee, M. Sugiyama, U.V. Luxburg, I. Guyon, R. Garnett (Eds.) Advances in Neural Information Processing Systems 292016.

[31] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, K. Kavukcuoglu, WaveNet: a generative model for raw audio arXiv, arXiv, DOI (2016) 15 pp.-15 pp.