SVD-based layout representation for lithographic hotspot detection

Yibo Huang[a+], Silin Chen[a+], Hao Shu[a], Jiahao Wang[b], Wenbo Xu[a], Ningmu Zou*[ac]
[a]School of Integrated Circuits, Nanjing University, 1520 Taihu Road, Suzhou, Jiangsu, China 215163; [b]School of Intelligent Software and Engineering, Nanjing University, 1520 Taihu Road, Suzhou, China 215163; [c]Interdisciplinary Research Center for Future Intelligent Chips (Chip-X), Nanjing University, 1520 Taihu Road, Suzhou, Jiangsu, China 215163; [+]Yibo Huang and Silin Chen contributed equally to this work; *Corresponding author: Ningmu Zou (nzou@nju.edu.cn)

ABSTRACT

With the continuous scaling of semiconductor manufacturing, lithographic hotspot detection has become crucial for improving manufacturing yield. Due to the extremely high spatial resolution of layout patterns, directly converting layout into raw images for deep-learning based hotspot detection leads to high computational complexity, so compact and informative layout representations are essential. Existing approaches have important limitations: the discrete cosine transform (DCT) tends to discard high-frequency details, while the squish pattern technique will introduce inconsistent data semantics across channels that impede neural network training. To address these challenges, this paper proposes a novel layout representation method based on singular value decomposition (SVD). By exploiting layout characteristics, the SVD-based approach enables significant lossless data reduction in deep learning training while preserving topological information from the original patterns which is critical for hotspot detection. On public ICCAD datasets, our method attains an average lossless data reduction rate of 98.8% compared to converting layout into raw images in model training. To validate the effectiveness of SVD representation, we integrate the reorganized SVD features and other representation tensors with ResNet18 to perform hotspot detection. Experimental results show that the SVD-based representation outperforms alternative representation methods. Moreover, with the simple ResNet18 architecture, the proposed SVD representation attains performance that matches or exceeds other complex state-of-the-art hotspot detection approaches. These findings indicate that the SVD method efficiently representing layout information while preserving the discriminative features necessary for robust and reliable hotspot detection.

Keywords: Lithographic Hotspot Detection, Layout Representation, Singular Value Decomposition, ResNet

1. INTRODUCTION

With the advancement of semiconductor manufacturing, the persistent scaling of transistor feature sizes and the increasing complexity of integrated circuits have presented significant challenges for design for manufacturability (DFM). Among various DFM tasks, lithographic layout hotspot detection aims to identify layout patterns that are likely to form printing failures such as breaks or bridges during lithography due to optical proximity effects (OPE), at an early stage of the design flow. It plays a crucial role in improving manufacturing yield.

Hotspot detection methods are primarily categorized into lithography simulation, pattern matching and deep learning (DL) based approaches[1]. Lithography simulation methods[2] model the lithography process with detailed mathematical and physical models and perform accurate computations to identify hotspots, but full-chip lithography simulation is computationally expensive and time-consuming, requiring extensive process and design-rule knowledge[3, 4]. Pattern-matching techniques[5-8] detect hotspots by comparing the layout against a predefined library of hotspot patterns. However, in advanced process nodes the diversity of failure modes and the complexity of layout images make it extremely challenging to construct a comprehensive and representative hotspot library, and impose stricter requirements on feature selection and matching strategies[9].

In recent years, DL techniques have been increasingly applied to DFM applications[10-12], bringing notable progress in layout hotspot detection[13-18]. DL techniques are capable of learning hidden relations between layout patterns and their defect characteristics, greatly improving detection accuracy. Current DL-based methods primarily convert the layout into an image and construct complex model architectures to address the hotspot detection problem. However, due to the extremely high spatial resolution of layout patterns (e.g. 1 nm), the generated layout image can be exceedingly large. Directly resizing these images leads to a severe loss of spatial detail. Consequently, an effective layout representation for dimensionality reduction is essential.

Ninth International Workshop on Advanced Patterning Solutions (IWAPS 2025), edited by Yayi Wei, Tianchun Ye, Proc. of SPIE Vol. 13991, 1399114 · © 2025 SPIE
0277-786X · doi: 10.1117/12.3093251

Existing layout representation methods, such as the Discrete Cosine Transform (DCT)[19] and the squish pattern technique[20, 21], suffer from significant limitations. The DCT-based approach, which retains only low-frequency coefficients of the layout, often fails to preserve high-frequency details critical to layout hotspot detection, such as right-angle corners or abrupt variations in metal-line spacing. The squish pattern method exploits the Manhattan geometry of layouts by encoding them as a binary presence matrix of layout elements and two numerical matrices for the width and height at each location. But the inconsistent data semantics across these channels may impede deep neural network training.

In this work, we propose a novel layout representation method based on singular value decomposition (SVD). The method leverages the characteristics of layout patterns to achieve efficient lossless data reduction while preserving key topological information from the original patterns, generating high-fidelity layout tensors for deep learning training. The main contributions of this paper are as follows:

1. We introduce SVD as a layout-representation technique that reduces data volume while preserving critical topological information of layout patterns.
2. We demonstrate that SVD can achieve high-rate lossless data reduction compared to converting layout into raw images in deep learning training and validate this empirically on public datasets.
3. We integrate the SVD-based representation with a simple ResNet18[22] backbone to achieve efficient and highly accurate hotspot detection, thus validates the effectiveness of SVD in layout representation.

The remainder of this paper is organized as follows. Section 2 defines the problems of hotspot detection and layout representation. Section 3 describes the proposed SVD method in detail, explains why it is efficient for layout representation, analyzes the high-fidelity preservation of topological information in the extracted features, and validates these features within a simple ResNet18 backbone in hotspot detection. Section 4 presents experimental results for data reduction and hotspot detection on public ICCAD datasets. Finally, Section 5 concludes the paper.

2. PRELIMINARY

2.1 Lithographic Hotspot Detection

A lithographic hotspot is a layout clip whose geometric features make it highly susceptible to manufacturing defects during photolithography. The hotspot detection problem can be defined as a binary classification task. Let TP, FP, FN and TN denote true positives, false positives, false negatives and true negatives respectively. The following metrics are used to evaluate the performance of a hotspot detector.

Definition 1 (Recall). Recall is defined as the fraction of ground-truth hotspots that are correctly detected.

$$Recall = \frac{TP}{TP + FN} \tag{1}$$

Definition 2 (False Alarm Rate). False Alarm Rate (FAR) refers to the fraction of non-hotspots incorrectly classified as hotspots.

$$FAR = \frac{FP}{TP + FN} \tag{2}$$

With these metrics, we formulate the hotspot detection problem as follows:

Problem 1 (Hotspot Detection). Given a set of layout clips, the goal is to develop a classification model that achieves a high recall while maintaining a reasonably low FAR, thereby ensuring that the maximum number of true defects are found without introducing an excessive number of false warnings.

2.2 Layout Representation

Raw layout data (e.g., GDSII) are high-resolution, Manhattan-geometry designs that are both spatially sparse and rich in small but critical geometric features. For DL models to process circuit layouts, the raw layout data must be converted into a suitable numerical tensor. A straightforward representation is converting the layout to a high-resolution image. However, large-scale layouts tend to be spatially sparse, resulting in significant computational overhead and an increased risk of overfitting when processed by DL-based models. Therefore, a layout representation g maps a high-dimensional layout clip l to a compact tensor T suitable for model learning utilizing layout properties:

$$g : l \mapsto T \in \mathbb{R}^{C \times H' \times W'} \tag{3}$$

An ideal layout representation method should achieve efficient data reduction to avoid computational overhead in model training, and more importantly, it must maintain high fidelity to the original layout. This means that topological features critical to hotspot detection must be effectively preserved in the resulting tensor T.

3. METHOD

3.1 Utilizing SVD for efficient layout representation

As illustrated in Fig. 1, singular value decomposition (SVD) is a theorem in linear algebra which states that a layout clip represented as a matrix $A \in \mathbb{R}^{m\times n}$ can be factorized as the product of a left orthogonal matrix $U \in \mathbb{R}^{m\times m}$, a diagonal matrix $\Sigma \in \mathbb{R}^{m\times n}$, and the transpose of a right orthogonal matrix $V \in \mathbb{R}^{n\times n}$, i.e.

$$A = U\Sigma V^T \tag{4}$$

The entries on the diagonal of Σ are called the singular values. The columns of U and the rows of V^T are referred to as the left and right singular vectors, respectively. Equivalently, the decomposition may be written as a sum of rank-one components:

$$A = \sum_{i=1}^{k} \sigma_i u_i v_i^T \tag{5}$$

where k is the number of non-zero singular values, σ_i is the i-th singular value, $u_i \in \mathbb{R}^m$ and $v_i \in \mathbb{R}^n$ are its corresponding left and right singular vectors, respectively. Furthermore, because the features derived from SVD are inherently correlated with the topology of the original layout, the matrices U and V (and their respective column vectors u_i and v_i) are designated as the layout's row and column information matrices (and vectors). The detailed explanation of these terms will be provided later in this work.

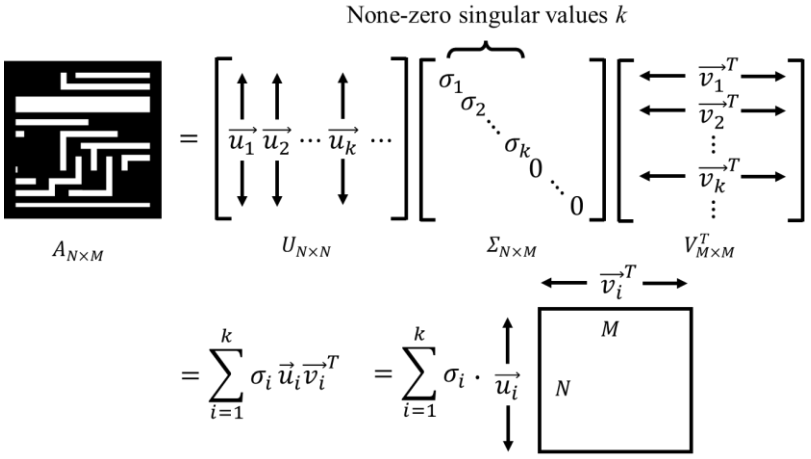

Fig. 1. An illustration of SVD for layout representation and its features preserving layout topological structure

After applying SVD to layout, lossless data reduction can be achieved by discarding zero singular values. We find that SVD is especially well-suited for layout representation and enables an extremely high lossless data reduction rate compared to converting layout into raw images in model training. Here is a brief explanation: first, the number of nonzero singular values of a matrix equals its rank as equation (6). Since U and V are orthogonal, multiplication by them does not change the rank of a matrix; therefore, for layout matrix A, $rank(A) = rank(\Sigma)$. The rank of a diagonal matrix Σ equals the number of its nonzero diagonal entries, i.e. the number of nonzero singular values k:

$$rank(A) = rank(U\Sigma V^T) = rank(\Sigma) = k \tag{6}$$

Second, layout images are low-rank as equation (7). The rank of a matrix is defined as the maximum number of linearly independent rows or columns. Because layout image contains many repeated rows and columns, these rows/columns are linearly dependent, making image rank is much smaller than its dimensions:

$$rank(A) \ll m, n \tag{7}$$

Consequently, most singular values of a layout matrix are zero, and only keeping a small number k of nonzero singular values and their associated singular vectors are sufficient to represent the layout matrix without loss of information.

Beyond lossless data reduction, SVD features also preserve the layout's topological information. The SVD explicitly decomposes the layout matrix into the sum of rank-one terms $\sigma_i u_i v_i^T$. Each term is a rank-one matrix formed by the outer product of a left and a right singular vector scaled by σ_i, and therefore represents a distinct component of the original layout, as shown in Fig. 1. This algebraic structure inherently embeds layout information within the singular vectors; specifically, u_i encapsulates the row-wise information while v_i^T summarizes the column-wise information for that particular component. The magnitude of the singular value σ_i reflects the contribution of the component to the original layout: large σ_i corresponds to components that capture overall structure, whereas smaller σ_i captures fine details. Therefore, U / u_i and V / v_i are naturally designated as row- and column-information embeddings of the layout; consequently, deep models trained on these SVD-derived features can effectively learn and exploit the topological characteristics of the original layout.

3.2 Hotspot detection with SVD features

To validate the effectiveness of SVD for layout representation, we apply the SVD features to the hotspot detection task and compare their performance with raw images and other layout-representation methods under a ResNet18 backbone. To enable the backbone network to fully exploit the SVD features, we reorganize them into a 2-channel feature tensor as illustrated in Fig. 2. Specifically, we truncate the top $k = 30$ singular values and sort them in descending order. For each selected singular value σ_i, we compute $\sqrt{\sigma_i} u_i$ and $\sqrt{\sigma_i} v_i$, then aggregate all left-side vectors $\{\sqrt{\sigma_i} u_i\}_{i=1}^{k}$ into one matrix and all right-side vectors $\{\sqrt{\sigma_i} v_i\}_{i=1}^{k}$ into another matrix; these two matrices form the two channels of the resulting tensor.

Each layout clip is processed by this SVD-based representation to produce a two-channel tensor, which is fed into the ResNet18 backbone for training. Through a hierarchy of convolutional layers, the backbone jointly captures row-wise and column-wise as well as overall and detailed layout features through the reconstructed tensor, and a final fully connected layer maps the learned features to a binary probability as the hotspot prediction. Because hotspot examples are much rarer than non-hotspots[23, 24], we employ the focal loss[25], which dynamically down-weights easy samples and forces the network to focus on hard, misclassified samples and the scarce hotspot samples. During inference, the class with the higher predicted probability is taken as the model decision.

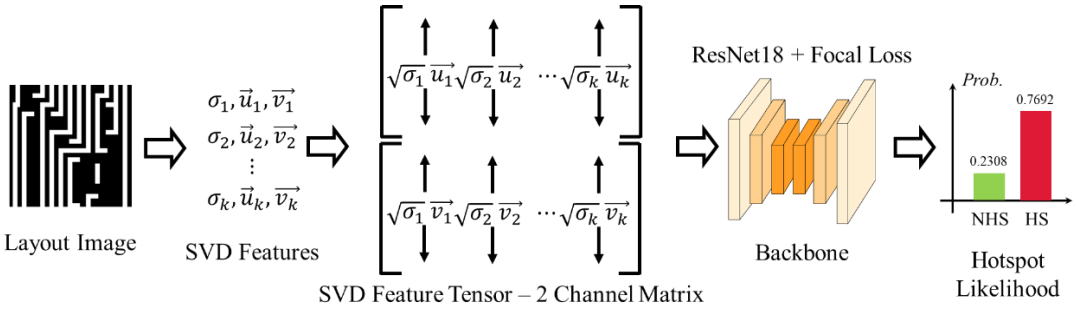

Fig. 2. SVD feature tensor generation for hotspot detection.

To further overcome the imbalance in the dataset, we perform data augmentation on hotspot samples. Owing to the symmetry of the lithographic illumination, certain geometric transformations (e.g., flips and rotations) that change the orientation of layout patterns do not affect their attributes[26]. Therefore, we apply operations listed in Fig. 3 to every hotspot sample in our training set to augment the data. Furthermore, we discover that these transforms correspond to simple operations on u and v. For example, reversing the order of singular-vector components or swapping the

corresponding u_i and v_i produces flips and rotations of the layout image, indicating that the layout's topological structure is preserved in the SVD features. Fig. 3 also illustrates the visual effects of these operations on the reconstructed patterns. In the figure, $u_i^{'}$ represents reverse the order of vector u_i, and the swapping of u_i and v_i is shown by switching their position.

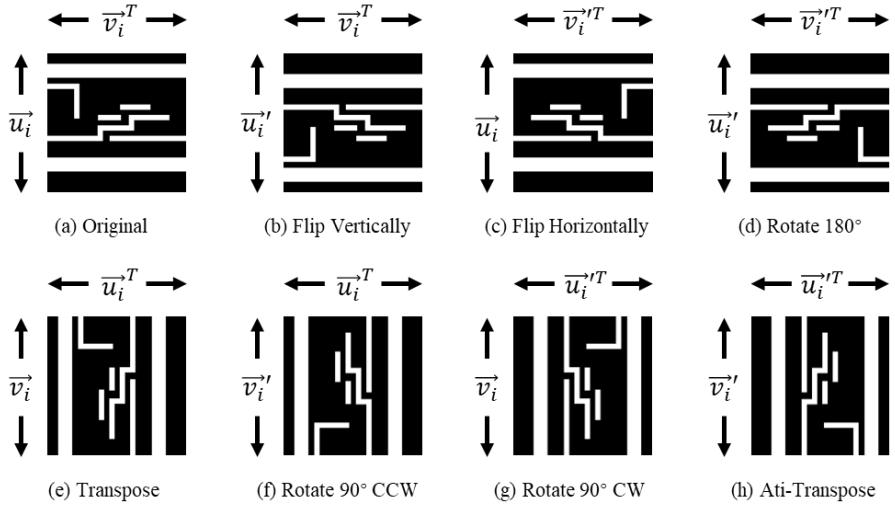

Fig. 3. Hotspot data augmentation methods with operations on *u* and *v*.

4. EXPERIMENTAL RESULT

4.1 Dataset

We evaluate the proposed SVD-based layout representation approach for data reduction and hotspot detection on the public ICCAD2012[23] and ICCAD2019[24] datasets. Each dataset contains series of layout clips with physical dimensions of 1200 nm×1200 nm and resolution of 1nm.

4.2 SVD data reduction result

To validate the efficiency of SVD for layout data reduction, we performed SVD on every layout in ICCAD2012 and ICCAD2019 and recorded the distribution of nonzero singular values k. As shown in Fig. 4, among the theoretical maximum of 1200 singular values only a very small subset is nonzero. This empirical low-rank property underlies the efficient lossless data reduction achievable by SVD. We measure the loss data reduction rate (R_{dr}) relative to storing layout as raw images for model training as

$$R_{dr} = 1 - \frac{(n+m+1)k}{nm} \tag{8}$$

where n and m are the width and height of the original image (here $n = m = 1200$), and k is the number of nonzero singular values retained. On the public ICCAD2012 and ICCAD2019 datasets this method achieves average lossless data reduction rates of 98.95% and 98.85%, respectively.

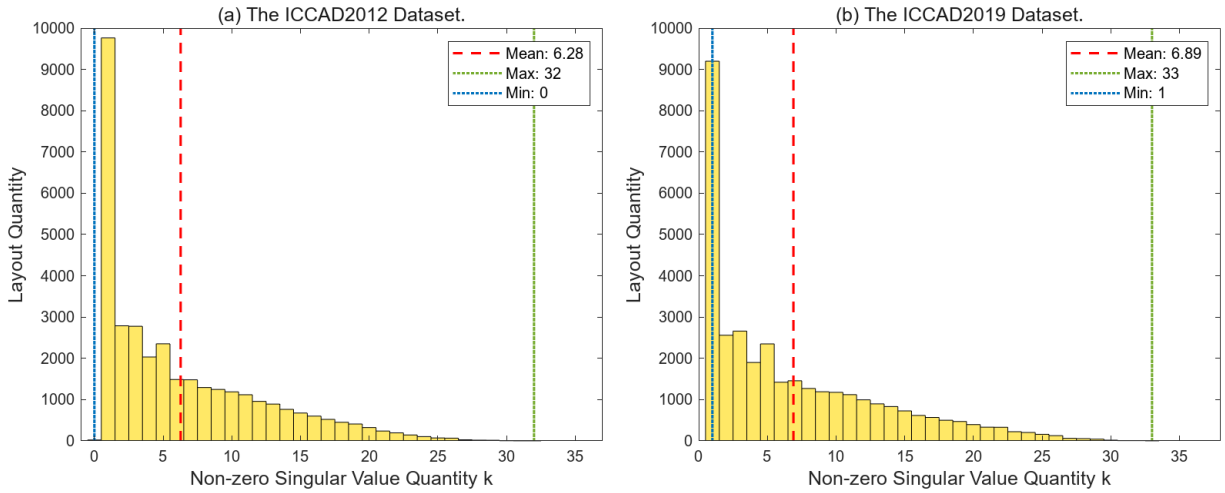

Fig. 4. Distribution of the number of non-zero singular values (k out of 1200) in the (a) ICCAD2012 and (b) ICCAD2019 dataset.

4.3 Hotspot detection setup

Hotspot detection experiments were carried out on a workstation equipped with an NVIDIA GeForce RTX 4080s GPU. In order to evaluate the quality of various layout representations, the tensors generated from these representation methods were trained on a ResNet18 backbone with focal loss to compare their detection performance and runtime. Implementation details for the compared representations are as follows:

- **DCT.** We follow the procedure in [19] to uniformly partition each layout into 12×12 regions. Each 100×100 patch is transformed by the Discrete Cosine Transform (DCT). Coefficients are read out via a zig-zag scan and the first 32 low-frequency coefficients are retained to produce a 32-channel 12×12 tensor.
- **Squish pattern.** Following the referenced squish-pattern method[20, 21], the layout is partitioned respecting the Manhattan geometry. Each partition is encoded as a binary presence matrix plus two numerical matrices recording local width and height, generating a 3-channel 128×128 tensor.
- **SVD.** SVD features are assembled into a 2-channel 1200×30 tensor as Section 3.

Because hotspot samples are far fewer than non-hotspot samples and geometric transforms like rotations and reflections do not alter a pattern's hotspot label, we augment the training set by applying the operations listed in Fig. 3 to every hotspot sample, resulting in an eightfold increase of hotspot instances. To accommodate different input tensor shapes produced by the three representation methods while keeping the model capacity comparable, we preserve the key model hyperparameters such as number of layers, kernel sizes, etc. and only adjust stride to achieve different pooling strategies as necessary.

4.4 Comparison between layout representation

Fig. 5 compares the detection performance and runtime of several layout-representation methods and direct raw-image processing when trained on a ResNet18 backbone with focal loss. Frame-per-second (FPS) denotes the number of layout images processed per second.

The results show that all three layout representation methods are over 10 times faster than processing raw 1200×1200 images, satisfying the efficiency requirements for backbone training. Regarding detection performance, on the ICCAD2012 dataset, our SVD-based representation attains the lowest FAR of only 0.1% while maintaining high recall. On the more challenging ICCAD2019 dataset, the SVD approach achieves the highest recall of 99.6% among all tested methods. These results indicate that SVD features offer advantages over other representation method, with the benefit becoming more pronounced on harder datasets.

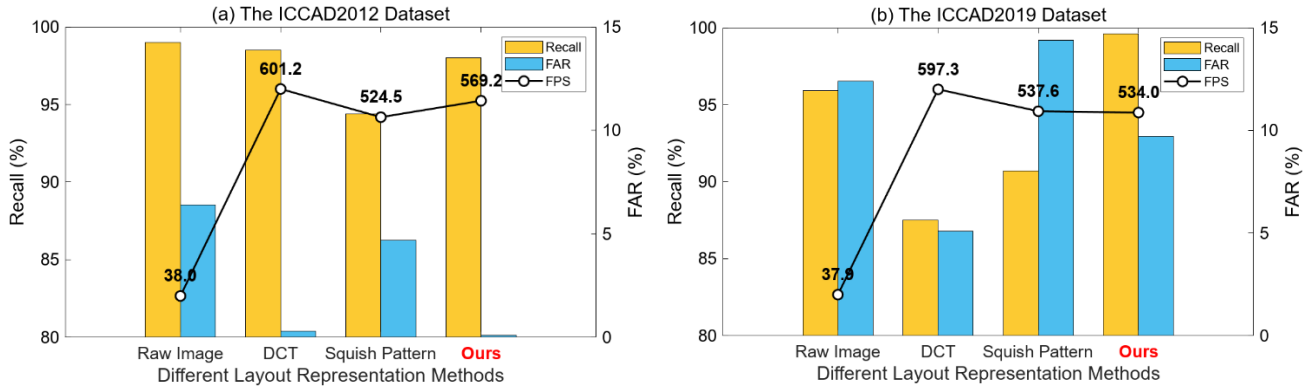

Fig. 5. Hotspot detection prformance comparison of different layout representation methods in the (a) ICCAD2012 and (b) ICCAD2019 dataset for hotspot detection.

4.5 Comparison with state-of-the-art

We further compare the simple SVD and ResNet18 detection framework against state-of-the-art (SOTA) methods, with results shown in Table 1. On ICCAD2012, our SVD method achieves a superior balance of recall and FAR. Compared to direct detection with ResNet in [26], it lowers the FAR by 3.8% at a comparable recall. On the more difficult ICCAD2019 dataset, the SVD method attains the highest recall, increasing the average recall of competing methods by 16.9% while maintaining a comparable FAR.

Under the simple backbone network, the SVD-based layout representation can also achieve detection performance that matches or surpasses more complex SOTA architectures. The experimental results validate that the proposed SVD layout representation efficiently reduces layout data while preserving critical topological information necessary for reliable hotspot detection.

Table 1. Comparison with state-of-the-art methods in the (a) ICCAD2012 and (b) ICCAD2019 dataset for hotspot detection.

(a) The ICCAD2012 Dataset

Methods	Recall (%)	FA (%)
IWAPS' 2022[26]	98.0	3.9
DATE' 2023[27]	96.2	6.4
DAC' 2024[17]	98.5	8.3
GLSVLSI' 2024[29]	88.3	0.2
TODAES' 2025[9]	98.1	4.0
TCAD' 2025[30]	**99.0**	5.8
SVD (Ours)	98.0	**0.1**

(b) The ICCAD2019 Dataset

Methods	Recall (%)	FA (%)
TODAES' 2022[16]	87.2	9.7
TCAD' 2020[28]	80.9	**2.5**
DATE' 2023[27]	91.6	8.6
GLSVLSI' 2024[29]	62.3	3.7
TCAD' 2025[30]	91.3	8.4
SVD (Ours)	**99.6**	9.7

5. CONCLUSION

In this paper, we introduced a novel SVD-based layout representation for lithographic hotspot detection to overcome the limitations of existing layout representation methods. By leveraging the inherent low-rank property of layout images, our method achieves an extremely high lossless data reduction rate compared to storing layout as raw images for model training. More importantly, the SVD features effectively preserve the critical topological information of the original layout patterns. Experimental results on public ICCAD datasets demonstrate that when integrated with a ResNet18 backbone, our SVD-based representation not only enhances computational efficiency but also achieves hotspot detection performance that is comparable to or surpasses that of more complex, state-of-the-art method. This validates that the proposed SVD representation is a highly effective approach for layout data reduction while maintaining the high-fidelity features essential for accurate hotspot detection.

ACKNOWLEDGEMENTS

Silin Chen was supported in part by Postgraduate Research Practice Innovation Program of Jiangsu Province (Grant Number: KYCX25_0394). Ningmu Zou was supported in part by the National Natural Science Foundation of China (62341408).

REFERENCES

[1] H. Yang, Y. Lin, B. Yu, and E. F. Y. Young, "Lithography hotspot detection: From shallow to deep learning," 2017 30th IEEE International System-on-Chip Conference (SOCC), 233-238(2017); https://doi.org/10.1109/SOCC.2017.8226047

[2] C. A. Mack, "Thirty years of lithography simulation," Proc. SPIE 5754, Optical Microlithography XVIII, 1-12(2005); https://doi.org/10.1117/12.601590

[3] J. Pan, X. Lin, J. Xu *et al.*, "Lithography Hotspot Detection Based on Heterogeneous Federated Learning With Local Adaptation and Feature Selection," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 43(5), 1484-1496(2024). https://doi.org/10.1109/TCAD.2023.3332841

[4] H. Sun, C. Jiang, X. Ye *et al.*, "Interpretable CNN-Based Lithographic Hotspot Detection Through Error Marker Learning," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 44(3), 1031-1044(2025). https://doi.org/10.1109/TCAD.2024.3468016

[5] Y. T. Yu, Y. C. Chan, S. Sinha *et al.*, "Accurate process-hotspot detection using critical design rule extraction," DAC Design Automation Conference 2012, 1163-1168(2012); https://doi.org/10.1145/2228360.2228576

[6] W. Y. Wen, J. C. Li, S. Y. Lin *et al.*, "A Fuzzy-Matching Model With Grid Reduction for Lithography Hotspot Detection," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 33(11), 1671-1680(2014). https://doi.org/10.1109/TCAD.2014.2351273

[7] W. C. Chang, and I. H. R. Jiang, "iClaire: A Fast and General Layout Pattern Classification Algorithm With Clip Shifting and Centroid Recreation," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 39(8), 1662-1673(2020). https://doi.org/10.1109/TCAD.2019.2917849

[8] X. He, Y. Wang, Z. Fu *et al.*, "A General Layout Pattern Clustering Using Geometric Matching-based Clip Relocation and Lower-bound Aided Optimization," ACM Trans. Des. Autom. Electron. Syst. 28(6), Article 90(2023). https://doi.org/10.1145/3610293

[9] C. Jiang, H. Sun, D. Feng *et al.*, "LithoExp: Explainable Two-stage CNN-based Lithographic Hotspot Detection with Layout Defect Localization," ACM Trans. Des. Autom. Electron. Syst. 30(3), Article 40(2025). https://doi.org/10.1145/3721129

[10] T. Chen, G. L. Zhang, B. Yu *et al.*, "Machine Learning in Advanced IC Design: A Methodological Survey," IEEE Design & Test. 40(1), 17-33(2023). https://doi.org/10.1109/MDAT.2022.3216799

[11] Z. Wang, Y. Shen, W. Zhao *et al.*, "DiffPattern: Layout Pattern Generation via Discrete Diffusion," 2023 60th ACM/IEEE Design Automation Conference (DAC), 1-6(2023); https://doi.org/10.1109/DAC56929.2023.10248009

[12] G. Zhou, B. Korrapati, G. R. Reddy *et al.*, "PatternPaint: Practical Layout Pattern Generation Using Diffusion-Based Inpainting," 2025 62nd ACM/IEEE Design Automation Conference (DAC), 1-7(2025); https://doi.org/10.1109/DAC63849.2025.11132857

[13] H. Zhang, B. Yu, and E. F. Y. Young, "Enabling online learning in lithography hotspot detection with information-theoretic feature optimization," 2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), 1-8(2016); https://doi.org/10.1145/2966986.2967032

[14] X. He, Y. Deng, S. Zhou *et al.*, "Lithography Hotspot Detection with FFT-based Feature Extraction and Imbalanced Learning Rate," ACM Trans. Des. Autom. Electron. Syst. 25(2), Article 15(2019). https://doi.org/10.1145/3372044

[15] Y. Xiao, M. Su, H. Yang *et al.*, "Low-Cost Lithography Hotspot Detection with Active Entropy Sampling and Model Calibration," 2021 58th ACM/IEEE Design Automation Conference (DAC), 907-912(2021); https://doi.org/10.1109/DAC18074.2021.9586273

[16] Y. Jiang, F. Yang, B. Yu *et al.*, "Efficient Layout Hotspot Detection via Neural Architecture Search," ACM Trans. Des. Autom. Electron. Syst. 27(6), Article 62(2022). https://doi.org/10.1145/3517130

[17] Y. Chen, Y. Wu, J. Wang *et al.*, "LLM-HD: Layout Language Model for Hotspot Detection with GDS Semantic Encoding," Proceedings of the 61st ACM/IEEE Design Automation Conference, Article 121(2024); https://doi.org/10.1145/3649329.3658479
[18] B. Zhu, S. Zheng, Y. Ma *et al.*, "Bridging Hotspot Detection and Mask Optimization via Domain-Crossing Masked Layout Modeling," ACM Trans. Des. Autom. Electron. Syst. 30(4), Article 54(2025). https://doi.org/10.1145/3728468
[19] H. Yang, J. Su, Y. Zou *et al.*, "Layout Hotspot Detection With Feature Tensor Generation and Deep Biased Learning," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 38(6), 1175-1187(2019). https://doi.org/10.1109/TCAD.2018.2837078
[20] F. E. Gennari and Y.-C. Lai, "Topology design using squish patterns," U.S. Patent 8,832,621 B1 (2014)
[21] H. Yang, P. Pathak, F. Gennari *et al.*, "Detecting Multi-Layer Layout Hotspots with Adaptive Squish Patterns," 2019 24th Asia and South Pacific Design Automation Conference (ASP-DAC), 1-6(2019); https://doi.org/10.1145/3287624.3288747
[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778(2016); https://doi.org/10.1109/CVPR.2016.90
[23] J. A. Torres, "ICCAD-2012 CAD contest in fuzzy pattern matching for physical verification and benchmark suite," 2012 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), 349-350(2012); https://doi.org/10.1145/2429384.2429457
[24] G. R. Reddy, K. Madkour, and Y. Makris, "Machine Learning-Based Hotspot Detection: Fallacies, Pitfalls and Marching Orders," 2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), 1-8(2019); https://doi.org/10.1109/ICCAD45719.2019.8942128
[25] T. Y. Lin, P. Goyal, R. Girshick *et al.*, "Focal Loss for Dense Object Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence. 42(2), 318-327(2020). https://doi.org/10.1109/TPAMI.2018.2858826
[26] M. Lin, F. Zeng, and Y. Shen, "Lithography hotspot detection with ResNet network," 2022 International Workshop on Advanced Patterning Solutions (IWAPS), 1-4(2022); https://doi.org/10.1109/IWAPS57146.2022.9972250
[27] Z. Chen, F. Yang, L. Shang, and X. Zeng, "Automated and Agile Design of Layout Hotspot Detector via Neural Architecture Search," 2023 Design, Automation & Test in Europe Conference & Exhibition (DATE), 1-6(2023); https://doi.org/10.23919/DATE56975.2023.10137142
[28] Y. Jiang, F. Yang, B. Yu *et al.*, "Efficient Layout Hotspot Detection via Binarized Residual Neural Network Ensemble," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 40(7), 1476-1488(2021). https://doi.org/10.1109/TCAD.2020.3015918
[29] C. Wang, Y. Fang, and S. Zhang, "Feature Fusion based Hotspot Detection with R-EfficientNet," Proceedings of the Great Lakes Symposium on VLSI 2024, 446–451(2024); https://doi.org/10.1145/3649476.3658707
[30] H. Yan, Y. Wang, P. Gao *et al.*, "A Lightweight Heterogeneous Graph Embedding Framework for Hotspot Detection," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 44(9), 3479-3489(2025). https://doi.org/10.1109/TCAD.2025.3543436