PAPER

An imbalance aware lithography hotspot detection method based on HDAM and pre-trained GoogLeNet

IOP Publishing Measurement Science and Technology
Meas. Sci. Technol. **32** (2021) 125008 (10pp) https://doi.org/10.1088/1361-6501/ac219a

An imbalance aware lithography hotspot detection method based on HDAM and pre-trained GoogLeNet

Kaibo Zhou[1], Kaifeng Zhang[1], Jie Liu[2], Yanan Liu[3], Shiyuan Liu[4], Guannan Cao[1] and Jinlong Zhu[4,*]

[1] Key Laboratory of Image Information Processing and Intelligent Control of Education Ministry of China, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, People's Republic of China
[2] School of Civil and Hydraulic Engineering, Huazhong University of Science and Technology, Wuhan 430074, People's Republic of China
[3] Center for Computational Electromagnetics, Department of Electrical and Computer Engineering, University of Illinois at Urbana−Champaign, Urbana, IL 61801-2991, United States of America
[4] State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan 430074, People's Republic of China

E-mail: jinlongzhu03@hust.edu.cn

Received 25 May 2021, revised 23 August 2021
Accepted for publication 26 August 2021
Published 10 September 2021

Abstract

Due to the continuous shrinkage of transistor size and the ever-increasing complexity of integrated circuit design layout, great challenges arise in optical lithography—any defect on the mask will be transferred to the silicon wafer, which may lead to severe defects such as open circuit and short circuit. These defects on masks are called hotspots. Before transferring the circuit layout on the mask to the silicon wafer, the entire mask must be inspected to accurately find out the hotspots before optical lithography. Although traditional lithography hotspot detection approaches, such as pattern matching and machine learning, have gained satisfactory results the performance of the model degrades when encountering problems such as complex layout and data imbalance. In this paper, a hotspot detection method based on hybrid data enhancement, data compression and pre-trained GoogLeNet is proposed to solve the aforementioned problems. Our study shows that the average recall rate can be up to 98.3%. Meanwhile, the false alarm is reduced and the F1-score is 63.5%. Experimental results show that the proposed method achieves better performance on the ICCAD 2012 contest benchmark compared to hotspot detection methods based on deep or representative machine learning.

Keywords: very large scale integration (VLSI), hotspot detection, hybrid data augment method, data compression, deep learning

(Some figures may appear in color only in the online journal)

1. Introduction

As the feature size of transistors continuously shrinks and the design layout of integrated circuit (IC) is becoming more and more complicated, the manufacturing precision and technical requirements of IC are increasing gradually [1]. The ICs defect detection becomes more and more challenging. In the optical lithography phase, most of the challenges arise from proximity effects caused by diffraction as light passes through a mask, and it may lead to severe defects such as open-circuit and short-circuit on the wafer. These defects on masks are called hotspots. Figure 1 presents four patterns with their lithographic

* Author to whom any correspondence should be addressed.

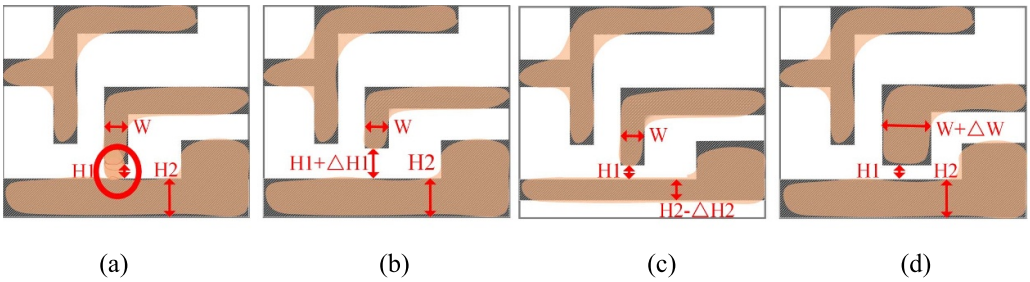

Figure 1. (a) A hotspot pattern and (b)–(d) non-hotspot patterns. Sub-figures (a)–(d) correspond to different photomasks (black parts indicate the mask patterns and brown patterns indicate the shape of lithography patterns).

simulation contours. The black parts are the mask before lithography, and brown domains represent the lithographic simulation contours. Due to small process margin in the lithography process, pattern (a) may cause bridges defects on the wafer after manufacturing (see the structures circled by a red circle). So, it is a hotspot sample [2]. Patterns (b)–(d) are non-hotspots due to the subtle differences. For example, (b) has a wider gap between two polygonal to avoid short circuit. For pattern (c), the diffraction of light is weaker because the decrease of width H2 reduces the light diffraction around the edge. Pattern (d) achieves the same effect by increasing width *W*. Therefore, it is necessary to implement defect detection on the circuit design layout followed by eliminating the fatal defects, so as to improve the IC quality rate and reduce the loss caused by the subsequent processes [3–5].

Generally, hotspot detection can be divided into several categories, including lithographic simulation-based [6], pattern matching-based [7] and machine learning-based [8, 9]. Lithography simulation is considered as the most standard one for lithography hotspot detection, but it is very challenging and time-consuming to perform lithography simulation on the full plate. The explosive growth of lithography model and design layout complexity has limited the application of lithography simulation-based hotspot detection. The pattern matching-based hotspot detection can find the hotspot contained in the hotspot library of a specific map, but it has no ability to detect the unknown hotspot, and it is a complex process to perfect a hotspot library. To shorten the detection time and solve the imperfection of the traditional methods, machine learning is still widely used in hotspot detection. Summarily, hotspot detection based on machine learning uses trained models to find out the missing hotspots, but it needs to extract features manually and frequently, which is prone to hotspot omission and misjudgment problems.

Recently, intelligent computing methods develop rapidly, and a series of excellent performance of deep network models, such as: Alex-Net, VGG-Net, SuperGraph, etc [10–12], have sprung up. These methods overcome the complex framework, detection performance problems, and the requirement of expert knowledge, and performing accurate classification tasks benefitting from high-efficiency-feature learning and high-nonlinear models [13–16]. Shin *et al* [17] proposed convolutional neural network (CNN)-based lithography hotspot detection method, and it outperformed previous conventional machine learning and fuzzy matching methods in terms of detection accuracy and false alarm reduction. Recently, Jiang *et al* [18] proposed a new deep learning architecture based on binarized neural networks to speed up the neural networks in hotspot detection and achieved a good trade-off by applying ensemble learning approaches. Yang *et al* [19] established a hotspot-detection-oriented neural network model, and it demonstrated that deep neural networks have potential to solve manufacturability problems as circuit layouts advance to extreme scale. Generally, the most direct way to improve the network performance is to increase the network depth and width, but blindly increasing the network will bring many problems: (a) there are too many parameters, and it is easy to produce overfitting if the training data set is limited; (b) the large network brings computational complexity, that is difficult to apply in practice. In consideration of these two issues, the pre-trained GoogLeNet is adopted to implement hotspot detection. It has inception blocks and a deeper structure than other CNN models. We optimize the parameters by fine-tuning the model by a small amount of training samples.

However, it is not enough to adopt a good network model, and there are still several aspects to consider: (a) it is a complex and costly process to obtain the layout design containing hotspots. The designed layout needs to be first manufactured into a silicon wafer, and then tested to verify whether it contains hotspots before the specific design of hot spots can be obtained. Therefore, the actual number of hotspot layout designs available is very limited. (b) For hotspot detection, the mask image size is much larger than the traditional classification task, which means that special processing of the mask image or change of the network structure is required [20]. We can take these two problems as breakthrough points and adopt corresponding methods to improve detection accuracy.

As mentioned above, although auxiliary samples can be generated through operations such as rotation, translation, and greyscale, the number of auxiliary samples generated is usually small and there is non-negligible amount of redundant data. It cannot be applied when the original samples are extremely scarce. For those reasons, generative adversarial network (GAN) model is introduced, which can generate large pseudo-sample data with the help of discriminator and generator. We mix the span, translation and GAN to generate auxiliary samples to solve the problem of data imbalance.

In view of the above discussion, we developed a hotspot detection flow based on data preprocessing and deep learning. A hybrid data augment method was carried out before training

to generate expanded samples which contrapose the problem of extremely unbalanced data distribution. Then, all the mask images are compressed and the spatial relationship is maintained to emphasize feature information. Finally, GoogLeNet network model is used to extract features and output classification results. The main contributions are as follows:

(a) A hybrid data enhancement method (HDAM) is proposed to augment the data of mask image data and solve the problem of extreme imbalance between hot and non-hot mode data.
(b) Compression algorithm is adopted to compress the original layout and save the feature information as much as possible.
(c) Adopted the pre-trained GoogLeNet network model and fine-tuned the model through transfer learning. It has deeper layers, stronger expression ability and lower error rate compared with other CNN models.

The rest of the paper is organized as follows: section 2 studies the theoretical basis of the proposed method. Section 3 provides a comprehensive study on different learning strategies including imbalance-aware processing and parameters. Section 4 lists the experimental results, followed by a conclusion and discussion in section 5.

2. Method

This section presents the theoretical basis of proposed hotspot detection method, which mainly contains three parts: hybrid data augment, image compression and model constructing.

2.1. Hybrid data augment approach

Due to the severe imbalance of data categories, for example, the ratio of hotspot and non-hotspot samples in ICCAD5 exceeds 100, the auxiliary samples generated by a single data enhancement method have repeatability. Therefore, it is necessary to adopt a hybrid data expansion method. In the field of image processing [21, 22], common methods include rotation, brightness adjustment, adding random noise, etc. These methods can generate auxiliary samples, and they can be mixed to generate auxiliary samples. The general expression is shown in equations (1) and (2), and P'' is the generated auxiliary samples [23]. The generated samples can help the network learn translation invariance, rotation invariance and other characteristics

$$P' = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} P + RN, \tag{1}$$

$$P'' = P_{\mathrm{m}} + (In - P_{\mathrm{m}}) \times (1 + \mathrm{percent}), \tag{2}$$

where θ denotes rotation angle, where θ values $\frac{\pi}{2}$, π or $\frac{3\pi}{2}$. P is original sample, RN is random noise, P_{m} is the average brightness of the layout, and percent is the regulator in the range [0,1]. In this way, the calculated average brightness is guaranteed to remain unchanged.

On this basis, GAN is introduced to generate the final enhanced samples. It can generate real-like samples, and the core components are generator and discriminator. GAN does not need to consider the model input, that means the input is noise image, and it can generate enough auxiliary samples. The specific process is shown in figure 2. The optimization objective of GAN can be defined as the binary cross-entropy loss:

$$L(G,D) = \min_G \max_D E_{x\sim P_r}[\log D(x)] + E_{z\sim P_z}[\log(1 - D(G(z)))], \tag{3}$$

where $D(x)$ and $D(G(z))$ denote the output of discriminator when the input is x and $G(z)$, respectively.

The basic process of GAN model is as follows. The network's input does not depend on any prior assumption, i.e. the input signal is random noise. Then, the false samples generated by the generator are compared with the real samples through the discriminator, and the results will be back propagated to optimize the generator and the discriminator. Finally, the generator will generate the real-like sample.

2.2. Data compression technique

The size of the mask layout is awfully enormous, and the hotspots are usually distributed in a certain area of the layout. Feature extraction of a complete layout image will cause a computational disaster, and the computational efficiency is low. Usually, the layout is split into pieces, which contain hotspot patterns or non-hotspot patterns. Nevertheless, the samples size is large. Hence, it is necessary to compress the sample images. The direct compression of the image will cause spatial distortion, and interpolation algorithm can help reduce the distortion to a minimum.

Lagrange interpolation is a common image scaling algorithm [24]. In general, the function $y = f(x)$ is defined in the interval $[a, b]$, with $n + 1$ distinct values $x_0, x_1, \ldots, x_n$, and the corresponding function value is $y_0, y_1, \ldots, y_n$, construct a polynomial through the $n + 1$ points with degree no more than n. It satisfies that $P_n(x_k) = y_k$, $k \in [0, n]$, and $P_n(x)$ is the interpolation function. Assume set $D_n = \{0, 1, \ldots, n\}$, and Lagrange basis function is $P_k(x) = \prod_{i \in B_k} \frac{x - x_i}{x_k - x_i}$ for $k \in D_n$. Then, the Lagrange interpolation polynomial is denoted as equation (4)

$$L_n(x) = \sum_{j=0}^{n} y_i P_j(x). \tag{4}$$

The Lagrange interpolation algorithm is used to scale the image, and the steps are shown in figure 3. The image size can be compressed by interpolation, the original spatial information of the layout can be retained, and the calculation efficiency is improved.

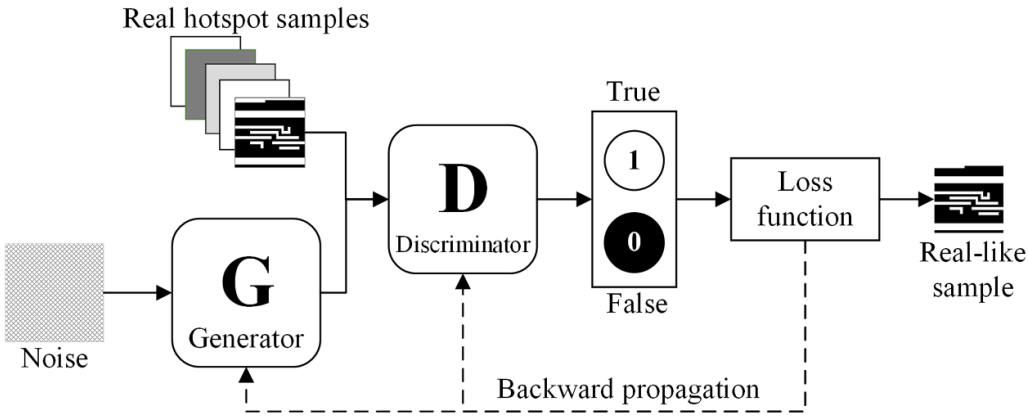

Figure 2. The schematic diagram of GANs.

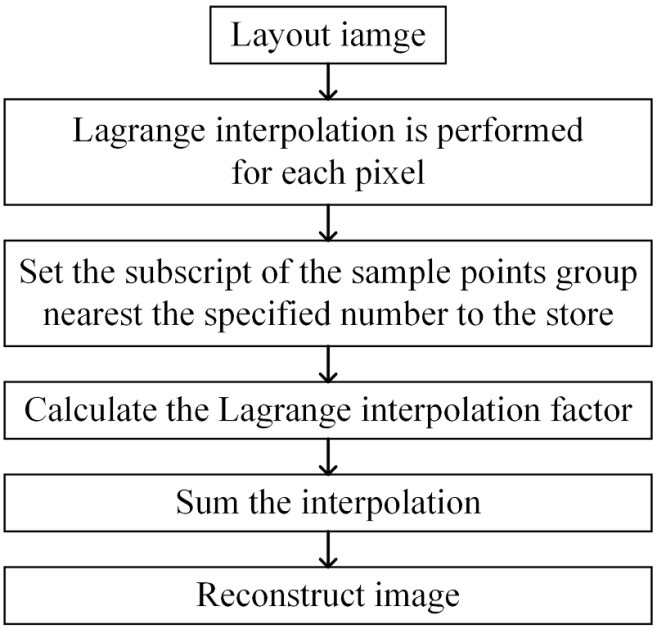

Figure 3. Layout compression based on Lagrange interpolation.

2.3. GoogLeNet

CNN is mainly composed of input layer, convolution layer, pooling layer, full connection layer and output layer [25–27]. The input data is usually a two-dimensional image. The convolution layer is the core of CNN, which is composed of filter and kernels, and it can extract the local features of the image by convolution kernel. Then, the new feature image is obtained by activation function. The pooling layer usually, also known as down sampling, is used to reduce the dimension of the feature image to keep local features unchanged and reduce calculation. The pool layer of the last layer outputs the advanced features of each image region, then it needs to combine these non-linear features in a simple way. The full connection layer fully connects the advanced features obtained through multiple convolution layers and multiple pooling layers to calculate the final predicted value. Finally, soft-max is used to classify the input image for classification task. It is vital to build a CNN with excellent performance for hotspot detection task.

Generally, the most direct way to improve the network performance is to increase the network depth and width, but blindly increasing the network will bring many problems: (a) there are too many parameters, and it is easy to produce overfitting if the training data set is limited; (b) the large network brings computational complexity, i.e. difficult to apply in practice. Meanwhile, the gradient dispersion problem might occur, and it is difficult to optimize the model. Training such deep CNN is time-consuming from scratch, and it requires a large amount of training data. In this case, it is desirable to use a pre-trained CNN based on large dataset to accomplish a similar task, which is known as transfer learning [28]. The structure of GoogLeNet is shown in figure 4. GoogLeNet applies inception, which assembles multiple convolution cores and pooling operations together to form a network module. When a network is designed, the entire network structure is assembled by modules as a unit, which reduces parameters and improves the expression of the network. Most strikingly, the network adopts average pooling instead of full connection layer, which can improve the accuracy. The pre-trained GoogLeNet model can be used for hotspot detection by fine-tuning for feature extraction.

In this research, the input of GoogLeNet is a layout image of size $224 \times 224 \times 3$. For hotspot detection, the last four layers need to be replaced when GoogLeNet is used for training, and the layers are replaced by dropout layer, full connection layer, soft-max layer and classification output layer. The dropout layer is used to help prevent over fitting, and the probability is set to 0.6. Gradient descent algorithm is used when training neural network.

3. Learning strategies

The flowchart of the proposed data augment and GoogLeNet-based deep learning method is shown in figure 5, that mainly contains four parts: data acquisition, data augment, data compression and GoogLeNet model constructing. The layout images with hotspot and non-hotspot are collected, and the hotspot data is extremely scarce compared to non-hotspot data.

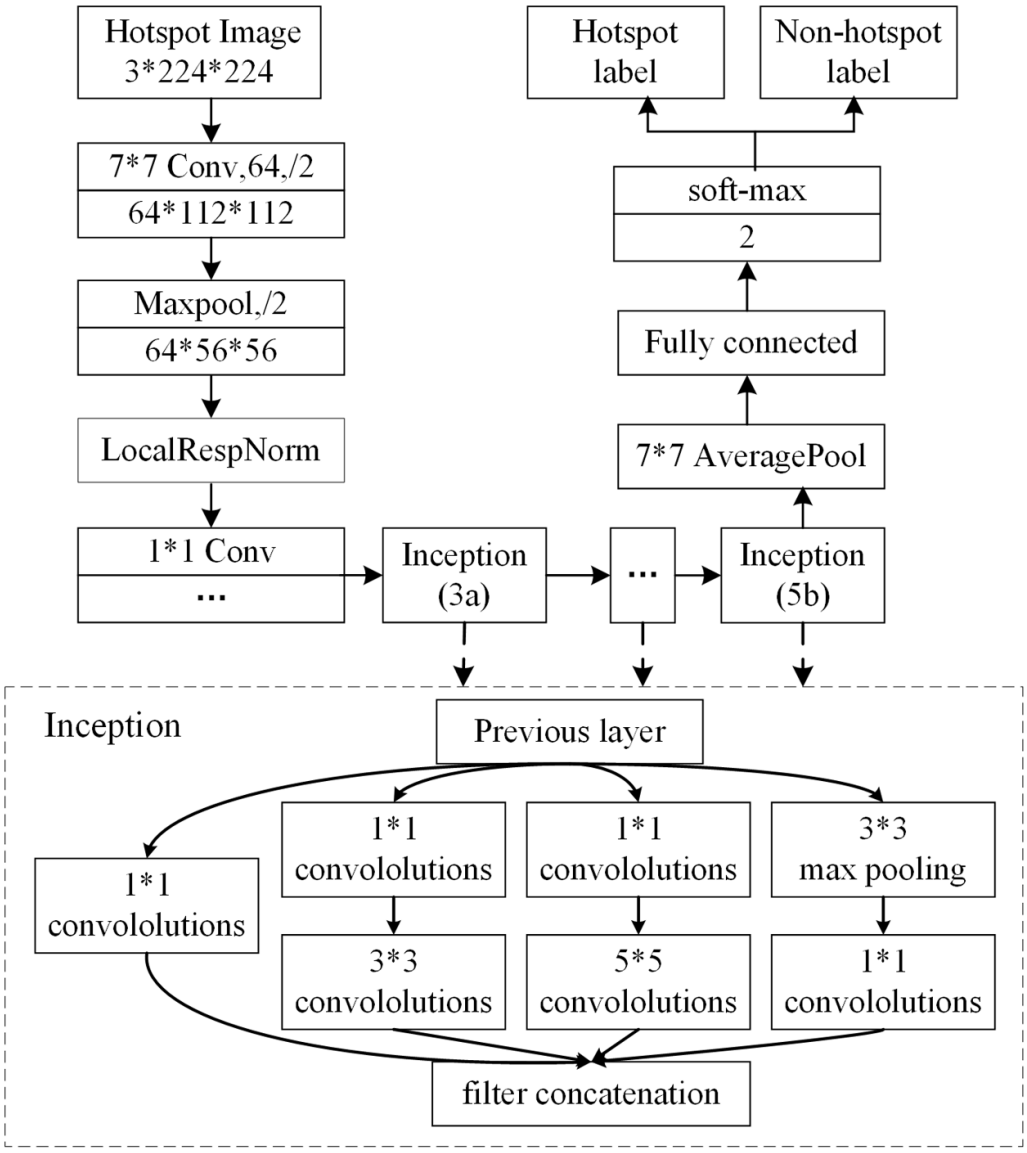

Figure 4. The structure of GoogLeNet model.

Each hotspot sample will be generated augmented images by a HDEM. Then original samples and auxiliary samples constitute the training set. Lagrange interpolation is used to solve the problem that the sample size is too large, and the final image size is 30 times smaller. All training samples and testing samples are processed in this step. Finally, the pre-trained GoogLeNet model is applied to train and test images to achieve detection of layout.

3.1. Data description and experiment preparation

To verify the effectiveness of the proposed method, this paper uses ICCAD2012 benchmark data set, which is shown in table 1. The data set contains image data obtained from hotspot status and non-hotspot status. The proposed method is to complete hotspot detection under conditions of extremely unbalanced data set. Since the unbalanced distribution of data sets may lead to non-hotspot category overfitting during the model training phase, the performance cannot be fully evaluated if only accuracy is used to measure it. For example, the non-hotspot sample distribution is extremely unbalanced in ICCAD-5. There are 41 hotspot samples and 19 327 non-hotspot samples in the test set. If all the samples are detected as non-hotspot, the accuracy is 99.79%. Therefore, it is incorrect to use only the accuracy rate as an indicator to evaluate the effectiveness of various methods. Here we use four metrics to evaluate the performance of the proposed method, namely, accuracy rate, precision rate, recall rate and F1-score.

The experimental computer is configured with Intel i7-8700k CPU and 32 G RAM, and GPU is NVIDIA GeForce GTX 1080 Ti.

3.2. The performance verification of HDAM

Due to the extremely unbalanced data distribution, the generalization ability of the training model is poor. Generally, two

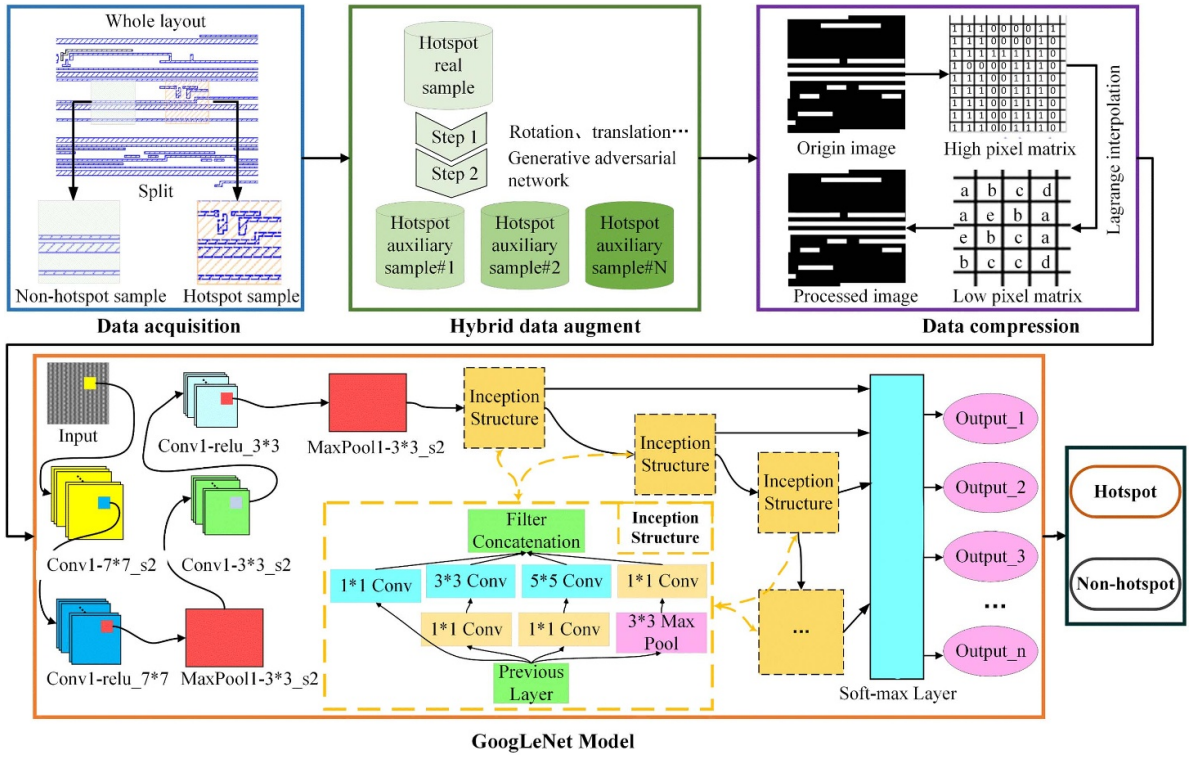

Figure 5. Flowchart of the proposed method.

Table 1. The description of hotspot and non-hotspot pattern for ICCAD 2012 benchmark.

Name	Technology (nm)	Hotspot number	Non-hotspot number	Area (μm^2)
ICCAD1	32 nm	226	319	12 516
ICCAD2	28 nm	498	4146	106 954
ICCAD3	28 nm	1808	3541	122 565
ICCAD4	28 nm	177	3386	82 010
ICCAD5	28 nm	41	2111	49 583

solutions are adopted to solve the problem of data imbalance: (a) data augment, that is, using the characteristic similarity of the existing samples to generate more new samples; (b) changing the categories' weight, giving different weights to different types of categories, so that the weighted loss of different categories is approximate. Since the proportion of hotspot samples and non-hotspot samples in the data set is too large, the changing the weight assignment has subjective factors, so the data augment method is adopted in this paper to solve the problem of data imbalance.

In this paper, HDAM is proposed to solve the problem of serious data imbalance. Due to the usage of a single data enhancement method, the generated auxiliary samples have great repeatability. Hybrid data augment method can generate similar samples, and these samples have common characteristics. In order to describe the reliability of auxiliary samples, it was measured by direct observation and maximum mean discrepancy (MMD). MMD is a measure of distance between probability distributions, and it can reflect the difference between the real sample and the generated sample. The calculation formula of the index is shown in equation (5)

$$\begin{aligned}\hat{d}_{\mathcal{H}}(X_s, X_t) &= \left\| \frac{1}{n_s}\sum_{i=1}^{n_s}\phi(X_i^s) - \frac{1}{n_t}\sum_{j=1}^{n_t}\phi(X_j^t) \right\|_{\mathcal{H}}^2 \\ &= \frac{1}{n_s^2}\sum_{i=1}^{n_s}\sum_{j=1}^{n_s}k(X_i^s, X_j^s) + \frac{1}{n_t^2}\sum_{i=1}^{n_t}\sum_{j=1}^{n_t}k(X_i^t, X_j^t) \\ &\quad - \frac{2}{n_s n_t}\sum_{i=1}^{n_s}\sum_{j=1}^{n_t}k(X_i^s, X_j^t), \end{aligned} \tag{5}$$

where X_s and X_t are the real data and the generated auxiliary sample, respectively. H indicates that this distance is measured by mapping the data into a regenerative Hilbert space. A lower MMD means that the two samples are closer together and that the quality of the generating image is higher.

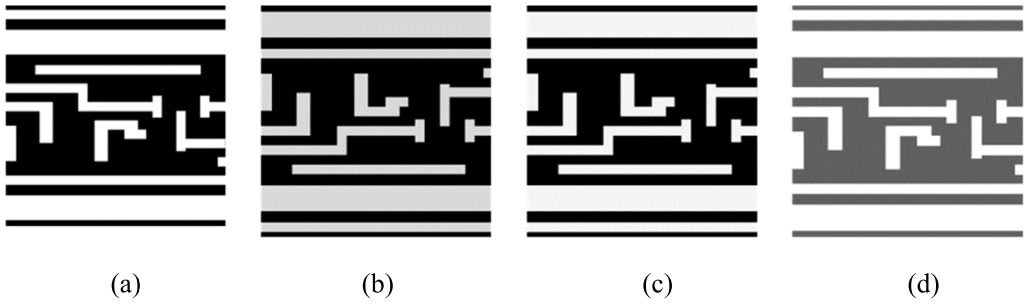

Figure 6. Real samples and auxiliary samples generated by HDAM.

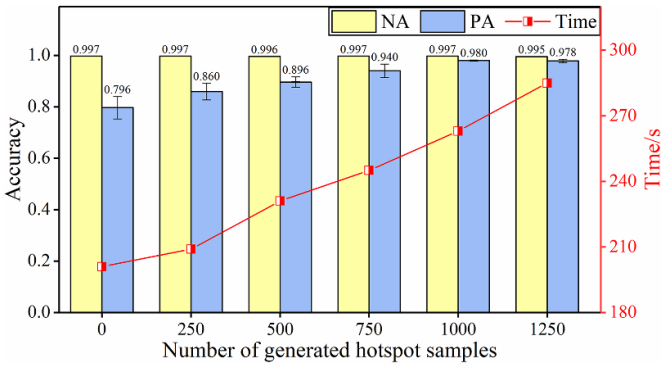

Figure 7. The effect of generating different number of hotspot samples on the results.

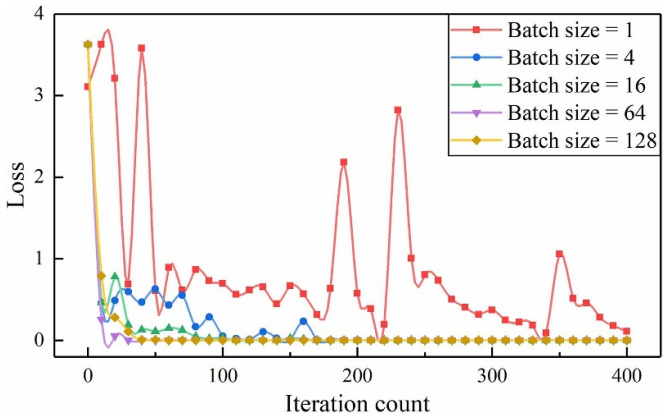

Figure 8. Effect of batch size.

The image generated by HDAM is shown in figure 6, where (a) is the real hotspot sample, (b)–(d) are the production auxiliary samples. The MMD value of HDAM is 0.1589. It can be seen from figure 6 that the generated auxiliary samples obtained are similar to the real samples by direct observation. The low value of MMD indicates that the generated samples are of high quality and have a similar distribution with the real samples. These auxiliary samples can help the network model to have anti-noise and anti-disturbance characteristics, so that the trained model can have better robustness.

The following will explore the influence of the number of generated samples on the experiment. If the number of auxiliary samples is too large, the training time will be increased greatly, and the weight of this category will increase, which results a loss of accuracy. If the number of samples generated is too small, the results will not be greatly improved. ICCAD-5 data set is used to discuss the influence of the number of generated samples on the data. The number of generated samples increases from 0 to 2000 by 200 each time. The result is shown in figure 7. Each experiment was repeated five times to reduce accidental error, and the results were measured by three indexes: positive accuracy (PA, a positive case is predicted to be positive), negative accuracy (NA, a negative case is predicted to be negative) and mean square error. It can be observed from the figure that the accuracy rate is continuously improved with the increase of the number of generated samples, which proves the effectiveness of the proposed mixed data augment method. When the number of samples generated is near 1000, the positive rate is the highest and the mean square error is low. At this point, the ratio of hotspot samples to non-hot samples is near 1:2. In subsequent experiments, HDAM is adopted to keep the ratio of hotspot samples to non-hot samples at 1:2.

3.3. Network parameter optimization

For deep networks such as GoogLeNet, different model training parameters affect the test results. This section mainly discusses the influence of different parameters on the results, and selects the best training parameters. The GoogLeNet network structure used in this paper has been proposed in section 2.3. This section mainly discusses two parameters: batch size and epoch, that affect the direction of the gradient. The impact of batch size and epoch on results is critical. When the amount of data is small, the whole data set can be used for learning, and the direction of gradient descent is more accurate. However, when the data set is large, the memory will be insufficient. Batch size is small, that means each correction is in the direction of the gradient of the respective sample, and makes the model difficult to converge. The larger the batch size is, the faster the same data will be processed, but the more epochs needed to achieve the same accuracy will be. Generally, the batch size has a saturation value, which will reduce the time and improve the efficiency. As shown in figure 8, ICCAD1 benchmark is taken as the research object to discuss the impact of different batch sizes and epochs on the results.

Table 2. Parameter settings.

Parameters	Value
Gradient algorithm	Adam
Mini batch size	64
Iterations	10
Learning rate	0.01
Momentum	0.9

According to figure 8, the model is self-learning when the batch size is equal to one. Each correction is in the direction of the gradient of the respective sample, and the loss is difficult to converge. Therefore, the batch size cannot be too small. Under the condition that the iteration count is identical, the model training speed is faster when the batch size is larger. When the batch size is 64, the loss converges fastest. When the batch size continues to increase and reaches saturation, the loss outweighs the gain by occupying computational memory. Hence, the batch size is set to 64 in subsequent experiments.

4. Experiment results

4.1. Comparison with different deep learning models

To verify that the proposed data preprocessing has a good generalization ability, it is proved that this method can achieve a good hotspot detection performance by collocating different network models. AlexNet has a deep network structure, using layered convolution layer, pooling layer to extract image features, and using dropout to suppress overfitting. Its performance is excellent in the field of image processing [29, 30]. Three sets of comparative tests are done, including: (a) AlexNet with HDAM, (b) GoogLeNet without HDAM and (c) GoogLeNet with HDAM. The parameters of all models are the optimal parameters, and the test conditions are the same to ensure the fairness of the experiment, and the parameter settings are shown in table 2. The results are shown in figure 9.

The proposed HDAM method is applied in two different network models, and recall rate is significantly improved, especially in ICCAD2, ICCAD4, and ICCAD5. The precision rate also increases by a certain amount by using HDAM. For the AlexNet and AlexNet with HDAM, the average recall accuracy is 93.9% and 97.1%, respectively. For the GoogLeNet and GoogLeNet with HDAM, the average recall accuracy is 95.2% and 98.31%, respectively, which is better than AlexNet. In terms of precision rate, the pre-trained GoogLeNet model with HDAM performs better than others, which also verified the effectiveness of the hybrid data augment method.

4.2. Comparison with existing hotspot detection methods

To demonstrate the effectiveness of the proposed method in the hotspot detection field. Here we compare our experiment

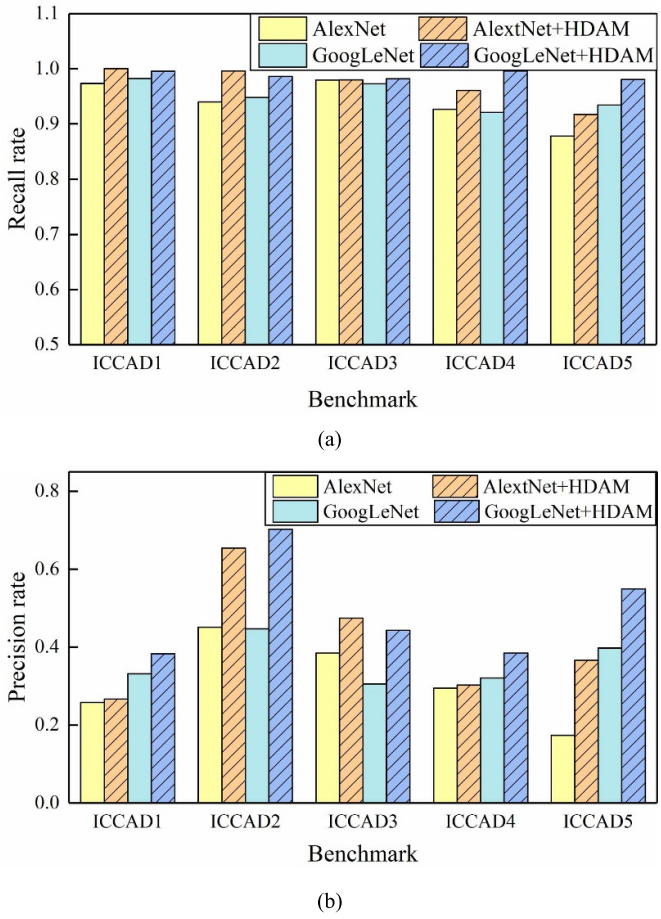

Figure 9. The detection accuracy of three models: (a) recall rate and (b) precision rate.

with a machine learning method, which adopts Principle Component Analysis and Support Vector Machine and Smooth Boosting, respectively [31]. This method is representative, and it achieves good results among many methods using machine learning. In addition, two additional deep-learning-based methods were compared to demonstrate the effectiveness of the proposed method. The 1st one is clustering and CNN [32], and the 2nd one uses HDAM and CNN to implement the detection task [3]. The main parameters settings of the comparative experiments are shown in table 3. The hotspot detection accuracy of the proposed method and comparison methods are illustrated in table 4.

As shown in table 4, the proposed method is significantly superior to other methods in terms of recall rate and precision. Thus, the proposed method based on HDAM, data compression and pre-trained GoogLeNet achieves a higher detection accuracy. The average recall rate, precision rate and F1-score of the five benchmark test sets are 98.3%, 47.5%, 63.5%, respectively. Thus, the proposed method based on HDAM, data compression and pre-trained GoogLeNet achieves a higher detection accuracy.

Table 3. Main parameters settings and algorithms of the comparative experiments.

Method	Parameter	Value
PCA and SVM	Maximum number of features	500
	Principal components	80
	Kernel function	Polynomial kernel
Clustering and CNN	Conv-pool layers (CPs) number	4
	PC activation	ReLU
	Full connected layer activation	Tanh
	Clustering algorithm	Density-based spatial clustering
Imbalance aware learning and CNN	CP layers number	87
	Imbalance aware learning algorithm	Random-mirror flipping and up-sampling

Table 4. Detection accuracy (%) on six detection tasks with different methods.

Data set	Method	Recall rate (%)	Precision rate (%)	F1-score (%)
ICCAD1	PCA and SVM	81.0	20.2	32.3
	Clustering and CNN	95.1	30.	46.3
	HDAM and CNN	100.0	17.9	30.3
	The proposed method	99.5	32.4	48.9
ICCAD2	PCA and SVM	81.1	3.9	7.4
	Clustering and CNN	99.5	19.0	31.9
	HDAM and CNN	98.7	84.6	91.1
	The proposed method	98.6	70.2	82.0
ICCAD3	PCA and SVM	87.0	5.4	10.2
	Clustering and CNN	98.9	7.8	14.5
	HDAM and CNN	98.0	36.0	52.7
	The proposed method	98.2	44.3	64.0
ICCAD4	PCA and SVM	80.5	4.7	8.9
	Clustering and CNN	97.6	6.8	12.7
	HDAM and CNN	94.5	35.4	51.5
	The proposed method	97.2	35.5	52.0
ICCAD5	PCA and SVM	84.1	8.6	15.6
	Clustering and CNN	97.9	15.6	26.9
	HDAM and CNN	95.1	8.96	16.4
	The proposed method	98.0	54.9	70.4
Average	PCA and SVM	82.7	8.6	14.9
	Clustering and CNN	97.8	16.0	26.5
	HDAM and CNN	97.3	36.6	48.4
	The proposed method	98.3	47.5	63.5

5. Conclusion

In this paper, we studied the feasibility of deep learning in lithography hot spot detection by focusing on two problems, i.e. data imbalance and large layout size. A HDEM is proposed to deal with data imbalance, which makes up for the deficiency of hot data. In view of the large layout size, i.e. the large sample size that affects the calculation efficiency of the model, data compression is adopted to reduce the data size. Finally, the trained GoogLeNet model was introduced and the ICCAD data set was used to fine-tune the model. Experimental results show that the proposed method is much more robust and performs better than existing deep learning approaches alongside representative machine learning approaches. The average recall rate and F1-score reached up to 98.3% and 62.9%, respectively. In conclusion, our study demonstrated that the difficulty of hotspot detection arising from data imbalance and large layout size can be elegantly resolved by our proposed strategy that consists of HDAM, data compression and pre-trained GoogLeNet.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Acknowledgments

This work was supported by the National Key Research and Development Program of China (No. 2020YFB1711203), PetroChina Innovation Foundation (No. 2020D-5007-0305), and Nondestructive Detection and Monitoring Technology for High Speed Transportation Facilities, Key Laboratory of Ministry of Industry and Information Technology (No. KL2019W003).

ORCID iDs

Kaibo Zhou https://orcid.org/0000-0003-0055-3193
Kaifeng Zhang https://orcid.org/0000-0003-1076-0209
Jie Liu https://orcid.org/0000-0002-0750-1030
Shiyuan Liu https://orcid.org/0000-0002-0756-1439
Jinlong Zhu https://orcid.org/0000-0002-5723-2879

References

[1] Reddy G R, Xanthopoulos C and Makris Y 2021 On improving hotspot detection through synthetic pattern-based database enhancement *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **9** 1–5
[2] Chen Y, Lin Y, Gai T, Su Y, Wei Y and Pan D Z 2019 Semi-supervised hotspot detection with self-paced multitask learning *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **39** 1511–23
[3] Yang H *et al* 2017 Imbalance aware lithography hotspot detection: a deep learning approach *J. Micro/Nanolithogr. MEMS MOEMS* **16** 033504
[4] Chen R, Zhong W, Yang H, Geng H, Yang F, Zeng X and Yu B 2020 Faster region-based hotspot detection *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **1** 1
[5] Liebmann L W 2001 Resolution enhancement techniques in optical lithography: it's not just a mask problem *Photomask and Next-Generation Lithography Mask Technology VIII* vol 4409 (International Society for Optics and Photonics) pp 23–32
[6] Guo J *et al* 2012 Improved tangent space based distance metric for accurate lithographic hotspot classification *DAC Design Automation Conf. 2012* (IEEE) pp 1169–74
[7] Chen K J *et al* 2017 Minimizing cluster number with clip shifting in hotspot pattern classification *Proc. of the 54th Annual Design Automation Conf. 2017* pp 1–6
[8] Lin Y, Xu X, Ou J and Pan D Z 2017 Machine learning for mask/wafer hotspot detection and mask synthesis *Proc. SPIE* **10451** 104510A
[9] Yu Y T *et al* 2015 Machine-learning-based hotspot detection using topological classification and critical feature extraction *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **34** 460–70
[10] Zhong W, Hu S, Ma Y, Yang H, Ma X and Yu B 2021 Deep learning-driven simultaneous layout decomposition and mask optimization *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **1** 1
[11] Yang C, Zhou K and Liu J 2021 SuperGraph: spatial-temporal graph-based feature extraction for rotating machinery diagnosis *IEEE Trans. Ind. Electron.* **1** 1
[12] Sathish K, Ramasubbareddy S and Govinda K 2020 Detection and localization of multiple objects using VGGNet and single shot detection *Emerging Research in Data Engineering Systems and Computer Communications* (Berlin: Springer) pp 427–39
[13] Deng L and Yu D 2014 Deep learning: methods and applications *Found. Trends Signal Proc.* **7** 197–387
[14] Zhou A *et al* 2020 On the defect detection for highly reflective rotary surface: an overview *Meas. Sci. Technol.* **32** 062001
[15] Kumar P R and Manash E B K 2019 Deep learning: a branch of machine learning *J. Phys.: Conf. Ser.* **1228** 012045
[16] Ezat W A, Dessouky M M and Ismail N A 2020 Multi-class image classification using deep learning algorithm *J. Phys.: Conf. Ser.* **1447** 012021
[17] Shin M and Lee J-H 2016 Accurate lithography hotspot detection using deep convolutional neural networks *J. Micro/Nanolithogr. MEMS MOEMS* **15** 043507
[18] Jiang Y *et al* 2020 Efficient layout hotspot detection via binarized residual neural network ensemble *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **40** 1476–88
[19] Yang H, Su J, Zou Y, Ma Y, Yu B and Young E F Y 2018 Layout hotspot detection with feature tensor generation and deep biased learning *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **38** 1175–87
[20] Jain A K 1981 Image data compression: a review *Proc. IEEE* **69** 349–89
[21] Liu J, Li Q, Zhang P, Zhang G and Liu M 2020 Unpaired domain transfer for data augment in face recognition *IEEE Access* **8** 39349–60
[22] Wang J *et al* 2020 Data augment method for machine fault diagnosis using conditional generative adversarial networks *Proc. Inst. Mech. Eng.* D **234** 2719–27
[23] Li J *et al* 2020 Data augment using deep convolutional generative adversarial networks for transient stability assessment of power systems *2020 39th Chinese Control Conf. (CCC)* (IEEE) pp 6135–40
[24] Zhang Z, Wang Q and Zhang Z 2021 Harmonic vector error analysis based on lagrange interpolation *IEEE Access* **9** 57464–74
[25] Chua L O and Roska T 1993 The CNN paradigm *IEEE Trans. Circuits Syst.* I **40** 147–56
[26] Wang S Y *et al* 2020 CNN-generated images are surprisingly easy to spot for now *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition* pp 8695–704
[27] Liu K *et al* 2020 Adversarial perturbation attacks on ML-based CAD: a case study on CNN-based lithographic hotspot detection *ACM Trans. Des. Autom. Electron. Syst.* **25** 1–31
[28] Weiss K, Khoshgoftaar T M and Wang D D 2016 A survey of transfer learning *J. Big Data* **3** 1–40
[29] Iandola F N *et al* 2016 SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and <0.5 MB model size (arXiv:1602.07360)
[30] Samir S, Emary E, El-Sayed K and Onsi H 2020 Optimization of a pre-trained AlexNet model for detecting and localizing image forgeries *Information* **11** 275
[31] Gao J R, Yu B and Pan D Z 2014 Accurate lithography hotspot detection based on PCA-SVM classifier with hierarchical data clustering *Proc. SPIE* **9053** 90530E
[32] Shin M and Lee J-H 2016 CNN based lithography hotspot detection *Int. J. Fuzzy Logic Intell. Syst.* **16** 208–15