

PGAU: Static IR Drop Analysis for Power Grid using Attention U-Net Architecture and Label Distribution Smoothing

Feng Guo*
Beijing University of Posts and
Telecommunications
Beijing, China
97guofeng@gmail.com

Jiawei Liu*
Beijing University of Posts and
Telecommunications
Beijing, China
liu_jiawei@bupt.edu.cn

Jianwang Zhai†
Beijing University of Posts and
Telecommunications
Beijing, China
zhaijw@bupt.edu.cn

Jingyu Jia
Beijing University of Posts and
Telecommunications
Beijing, China
jiayingyu@bupt.edu.cn

Kang Zhao
Beijing University of Posts and
Telecommunications
Beijing, China
zhaokang@bupt.edu.cn

Chuan Shi†
Beijing University of Posts and
Telecommunications
Beijing, China
shichuan@bupt.edu.cn

ABSTRACT

As feature sizes shrink, the on-chip power grid (PG) faces serious power integrity issues, and static IR drop analysis becomes critical for PG design and optimization. Many machine learning (ML) based methods have been proposed to address the inefficiencies of traditional numerical methods. However, many previous works have ignored the problems of feature confusion and imbalance IR drop distribution. In this work, we propose novel feature augmentation and selection methods to solve the feature confusion problem and use the label distribution smoothing (LDS) technique to handle unbalanced labels. Importantly, we design a static IR drop analysis model for PG using the Attention U-Net architecture (PGAU). Furthermore, two real-world datasets are used for evaluation. Experiments show that our model outperforms baselines, with a 2.6% improvement in the correlation coefficient (CC) and a 22.2% reduction in the mean absolute error (MAE). Moreover, our model is highly transferable and performs better against never-before-seen designs.

CCS CONCEPTS

• **Hardware** → **Physical design (EDA), Physical verification.**

KEYWORDS

Power grid, IR drop, Attention, U-Net

ACM Reference Format:

Feng Guo, Jiawei Liu, Jianwang Zhai, Jingyu Jia, Kang Zhao, and Chuan Shi. 2024. PGAU: Static IR Drop Analysis for Power Grid using Attention U-Net Architecture and Label Distribution Smoothing. In *Great Lakes Symposium on VLSI 2024 (GLSVLSI '24)*, June 12–14, 2024, Clearwater, FL, USA. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3649476.3658711>

*Co-first authors.

†Corresponding authors.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GLSVLSI '24, June 12–14, 2024, Clearwater, FL, USA

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-0605-9/24/06...\$15.00

<https://doi.org/10.1145/3649476.3658711>

1 INTRODUCTION

In the field of integrated circuit (IC) design, the on-chip power grid (PG) plays a critical role in transferring voltage and current to each working instance [1, 2]. However, parasitics in the PG introduce voltage drops (i.e., IR drops) between power pads and cells, potentially affecting the chip performance and functionality. Therefore, ensuring that the worst-case IR drop values are within specified limits is essential, which prevents detrimental effects on the chip. As chip complexity increases, IR drop analysis and optimization become more challenging. Traditional numerical methods accomplish the static IR drop analysis in the physical verification stage by solving large-scale sparse linear systems of equations [2]. However, these methods are very time-consuming in industrial-scale designs, often requiring hours or days. Therefore, accelerating analysis and providing early IR drop estimates open the door to optimizing the design cycle.

Methods for accelerating IR drop analysis with machine learning (ML) have emerged in recent years with superior accuracy and efficiency, compared with traditional equation solving. For example, XGBIR [3] constructs a predictor for static IR drop in PG based on XGBoost [4]. It accurately addresses IR drop variations across different scenarios by leveraging features extracted from the PG. However, as an instance-level predictor, it is difficult to model the local consistency of the IR Drop distribution. PowerNet [5] utilizes a convolutional neural network (CNN) to predict full-chip IR drop, employing instance-level power maps as features. Nonetheless, its assumption of uniform resistance from each instance to the power pin may not be suitable for real designs with irregular grid densities. IREDGe [6] introduces a U-Net-based EDGe network that transforms an input power image into a corresponding output static IR drop image. Both images represent contour maps over the identical die area, focusing on static IR drop analysis. MAVIREC [7] employs a 3D U-Net model for IR drop prediction to capture detailed, fine-grained on-chip information. While its ML strategy primarily emphasizes dynamic IR drop problems, it also applies to static IR drop analysis.

Although the aforementioned methods can be applied to static IR drop prediction, we observe that these methods overlook two key issues: feature confusion, reflected in the diversity and complexity of features provided by the original data, and imbalanced label distribution, which shows large differences in IR drop distribution on

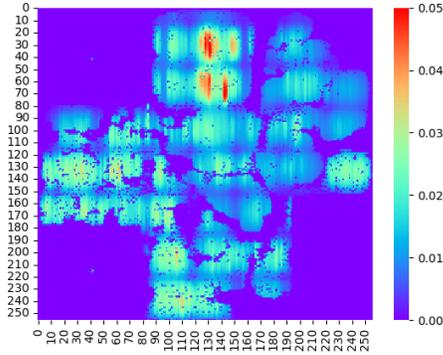


Figure 1: An example of IR drop distribution in PG, exhibiting locally consistent and overall imbalanced characteristics.

the PG. Limited by the size of datasets and the number of features, previous works [3, 5–7] ignored these problems. With the development of electronic design automation (EDA) tools, more and more features describing PG are extracted from netlists. This seems to be beneficial to learn more about PG quantification, but many of these features are of low quality and will cause adverse effects on IR drop analysis. Therefore, to achieve high-quality predictions, we should not blindly utilize all raw features but rather selectively choose or even construct more valuable composite features. Another problem comes from the imbalanced distribution of IR drop labels. Due to limitations of on-chip power and dark-silicon trends, the number of hotspots is relatively small and may occur anywhere in the chip [2], and the IR drop distribution is locally consistent and overall imbalanced. As shown in Figure 1, the hotspots are concentrated and few, and there are generally more normal nodes than hotspot nodes. This may lead to ambiguous hotspot detection and larger errors and thus is not suitable for training ML models.

To address the above problems, we make targeted improvements and innovations in this work. To solve the problem of feature confusion, we do a series of works on feature construction and selection. After obtaining the raw data from PG, we make feature augmentation to get three combined features to represent the information of combined power and neighboring information and perform feature importance analysis to eliminate less important and inefficient features. Importantly, we utilize the attention gate based on the U-Net architecture for enhancing model expressiveness. The attention mechanism focuses on the hotspots parts of the IR drop and the relationship between various PGs. Moreover, the IR drop distribution in the PG dataset is highly imbalanced, showing a skewed distribution. We use the label distribution smoothing (LDS) technique to re-weight the loss function for regression on imbalanced IR drop. Our method has been evaluated in a series of experiments on two different datasets, and its effectiveness and superiority have been verified.

The main contributions of our work are as follows:

- This work focuses for the first time on the complex and redundant features of PG and imbalanced IR drop distribution.
- To solve the problem of feature confusion, this work extracts more effective combined features from the original features, analyzes the importance, and filters low-quality features.
- This work proposes a customized universal U-Net-based ML model for accurate static IR drop prediction, using an attention

gate (AG) to capture the special area (e.g., hotspots) of IR drop in the PGs.

- To solve the problem of imbalanced IR drop distribution, this work adopts label distribution smoothing (LDS) to re-weight the loss function and improve the model’s performance.
- Experiments on the CircuitNet and ICCAD2023 datasets show the superiority of our model, surpassing the baselines with a 2.6% improvement in CC and a 22.2% reduction in MAE, and better model transferability.

2 PRELIMINARIES

2.1 Power Grid and IR Drop

In very large-scale IC development, the design of PG involves the analysis and optimization of the on-chip metal network that distributes power on the chip. As with all engineering, there are trade-offs involved - the network should have adequate and reliable performance, but not use more resources than needed [1]. Due to aggressive technology scaling, on-chip power density has been on an increasing trend over time [2]. In deep submicron technology, feature sizes are shrinking while power consumption per device is increasing. This results in high currents in the power and ground metal networks, which reduces performance and reliability. A robust PG is critical to ensuring the reliable operation of chip circuits.

The analysis of static IR drop holds significant importance as it serves as a crucial step in assessing the power integrity of the chip. Owing to the parasitics (a nonideal element or effect in a circuit) in the PG, specifically the partial voltage arising from the resistance of metal wires, voltage drops are induced between the power pads and the cells in the design as current travels through the PG. Large IR drops in the design can impact the chip’s performance and even compromise its functionality in extreme cases. Therefore, it is essential to implement checks to verify whether the IR drop values at the nodes of the PG connected to the instances (current sources) are within specified limits. Traditionally, determining the voltage at each node in PG involves solving a set of linear equations expressed as $GV = J$, where G represents the conductance matrix, V is the unknown vector of voltages, and J is the vector of currents. Nevertheless, solving this equation system becomes computationally prohibitive when dealing with millions or more PG nodes, which gives birth to ML methods, such as those in literature [4–7].

2.2 Image Segmentation and U-Net Architecture

To perform IR drop prediction, we need to analyze the chip layout. One commonly used ML technique is image segmentation, which achieves pixel-level prediction by semantically understanding different regions or layers in the chip layout. By converting PG to image and finely partitioning image pixels, we can predict the instance IR drop values for each pixel.

Recently, CNN [8] has significantly advanced image recognition and has become instrumental in the evolution of image segmentation, where the U-Net architecture [9] is one of the most widely used CNNs with an encoder-decoder structure. The encoder comprises convolutional and pooling layers for feature extraction and compression. It increases the receptive field, performs feature extraction over a larger image area, and enables multi-scale feature fusion. Conversely, the decoder upsamples the low-resolution feature mapping from the encoder output to the original resolution

through upsampling and inverse convolutional layers and produces segmentation results. The U-shaped architecture promotes the propagation of contextual information, allowing for object segmentation by leveraging context from a more extensive overlapping region. Moreover, U-Net incorporates skip connections. The feature map from the corresponding layer in the contraction path is clipped and concatenated into the upsampled feature map. It establishes links between specific encoder layers and corresponding decoder layers, thereby mitigating information loss and segmentation inaccuracies throughout the process [10]. In this work, we use the attention mechanism to further enhance the modeling ability of U-Net architecture for PG.

2.3 Problem Formulation

This work converts static IR drop analysis into an ML prediction task, aiming to solve the problem of feature confusion and distribution imbalance of IR drop labels in PG, and is ultimately reflected in the prediction of IR drop value and detailed distribution improvement in the PG.

In the context of this research, we adopt a perspective by treating the PG as a 2D spatial image, where in each PG design is translated into a corresponding data matrix. This conceptualization allows us to leverage the structural characteristics of the PG in a format akin to a multi-channel 2D image. To predict IR drops, as for the input features, we refer to each $w \times h$ data matrix representing different features as a feature map. To specifically address the prediction of IR drops within this spatial representation, we employ feature maps as input representations. Each feature map, denoted as P_{map_i} , corresponds to a $w \times h$ data matrix, essentially serving as a spatial representation of the inherent properties of the PG. The IR Drop of instances in the entire PG is also converted to a data matrix and represented as y . The target ML model F tries to give the closest prediction F^* on y based on all n different feature maps $\{P_{map_1}, \dots, P_{map_n}\}$. The concrete abstract formula is as follows:

$$F : \{P_{map_1} \in \mathbf{R}^{w \times h}, \dots, P_{map_n} \in \mathbf{R}^{w \times h}\} \rightarrow \mathbf{R}^{w \times h}, \quad (1)$$

where Equation (1) describes the mapping function F , which takes a set of n feature maps $\{P_{map_1}, \dots, P_{map_n}\}$ as input and produces a predicted output in the form of a $w \times h$ matrix.

$$F^* = \arg \min_F \text{Loss} \left(F \left(\{P_{map_1}, \dots, P_{map_n}\} \right), y \right), \quad (2)$$

where Equation (2) further refines the predictive model by introducing a loss function Loss , and the objective is to minimize this loss with respect to the parameters of the model F . The argument F^* represents the optimized model that minimizes the loss, thereby providing the most accurate prediction for the target variable y based on the input feature maps.

3 METHODOLOGIES

3.1 Overall

We focus on the static IR drop analysis task of PG, and Figure 2 illustrates our ML framework for IR drop prediction. To solve the problem of feature confusion and extract more effective PG information, we perform feature augmentation by combining important features filtered by XGBoost, and then construct feature maps. For more precisely capturing the PG feature information, especially the

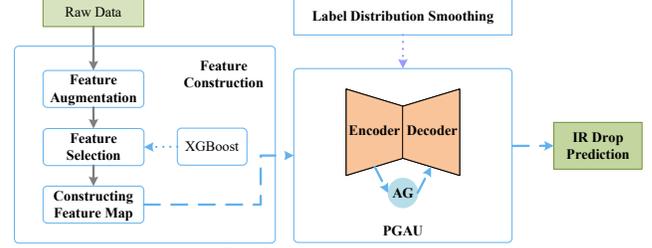


Figure 2: The framework of our method.

hotspots, we apply the attention mechanism for the first time in this domain based on the U-Net architecture. To handle the imbalanced IR drop distribution, we apply label distribution smoothing (LDS) on the loss function when training the PGAU model.

3.2 Feature Construction

3.2.1 Feature Augmentation.

Due to the problem of feature confusion described in Section 1, we try to extract more convincing features from the raw features, which are shown in Table 1. It can be clearly seen that the raw features provided by the PG are diverse and complex.

Table 1: Raw features of PG

Feature	Symbol
The cell type to which the instance belongs.	C_{inst}
Instance's equivalent resistance to all power pads.	R_{odd}
Instance's equivalent resistance to all ground pads.	R_{gnd}
Instance's loop resistance, i.e., $R_{odd} + R_{gnd}$	R_{loop}
Instance's minimum resistance to nearest power pad.	mR_{odd}
Instance's minimum resistance to nearest ground pad.	mR_{oss}
Instance's operating frequency.	f_{req}
Instance's toggle rate per unit cycle.	r_{tog}
Instance's leakage power.	$P_{leakage}$
Instance's switching power.	$P_{switching}$
Instance's internal power.	$P_{internal}$
Instance's total power.	P_{all}
The ideal supply voltage of power net.	$P/G - volt$
The power net to which the instance is connected.	$P/G - domain$
Power pin.	$P/G - pin$

Since the IR drop is directly affected by power consumption, we focus on extracting new features related to power. Upon receiving the raw features, we construct three new composite features to make feature augmentation: Composite Power, Neighbor Count, and Neighbor Power. Based on the power type in Table 1, we can generate more detailed power information. Composite power $P_{composite}$ [5] reflects the overall power dissipated by one instance and is scaled by the toggle rate of the instance, which is defined by:

$$P_{composite} = (P_{internal} + P_{switching}) r_{tog} + P_{leakage}, \quad (3)$$

where $P_{internal}$ is internal power, $P_{switching}$ is switching power, $P_{leakage}$ is leakage power and r_{tog} is toggle rate of the instance.

Neighbor power $P_{neighbor}$ and neighbor count $Count_{neighbor}$ are inspired by [3], which first put forward the neighbor voltage (i.e., $V2N$). $Count_{neighbor}$ is defined as the number n of instances within a specific radius of the target instance. $P_{neighbor}$ describes the power effects of these n nearest instances:

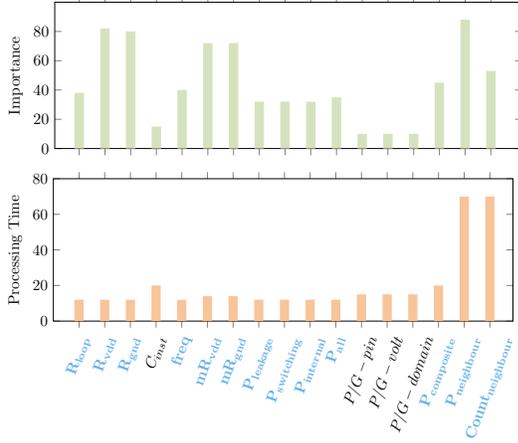


Figure 3: Results of feature analysis and selection.

$$P_{\text{neighbor}} = \frac{1}{n} \sum_i^n (P_{\text{all}})_i. \quad (4)$$

According to this method, we get the correlation information between the instances, which is important for IR drop analysis. The extraction of neighbor information conforms to the locality principle of static IR drop, which has also been confirmed by relevant literature.

3.2.2 Feature Selection.

For each PG, we refrain from considering the full range of potential features that might appear relevant, as this would result in an overly complex and over-fitted model. Our approach involves feature selection, utilizing XGBoost [4] to assess and rank the significance and process time of all features. The result is presented in Figure 3, where features with blue fonts are selected. Although the two features $\text{Count}_{\text{neighbor}}$ and P_{neighbor} are more time-consuming, we chose these two features to obtain better prediction accuracy.

3.2.3 Constructing 2D Spatial Feature Maps.

We treat PG as a multi-channel image and then we construct feature maps. Creating 2D spatial features involves a process of spatial location of per-instance features. Based on the row height from Library Exchange Format (LEF), w and l , a design of size $W_c \times L_c$ translates to an image of $W (= W_c/w) \times L (= L_c/l)$ pixels. In other words, the coordinates of each node x_n and y_n will be translated to $x = x_n/w$ and $y = y_n/l$. We choose a feature map size of 256×256 for fine-grained modeling. In this way, every node is planted into the 256×256 grid. If there are two nodes whose coordinates are particularly close, they are placed in the same grid position. Paddings around the layout (if existed) need to be subtracted.

3.3 Encoder-Decoder Architecture with Attention Mechanism

Referring to the equation Equation (1) in Section 2.3, we designed the model F specifically. To improve the ability to focus on hotspots, we apply the attention mechanism for the first time to IR drop prediction. We achieve this by using the attention gate (AG), which is a unit that trims features that are less relevant to the current task. The AG is used to focus on IR drop features in regions of higher variability, especially IR drop hotspots.

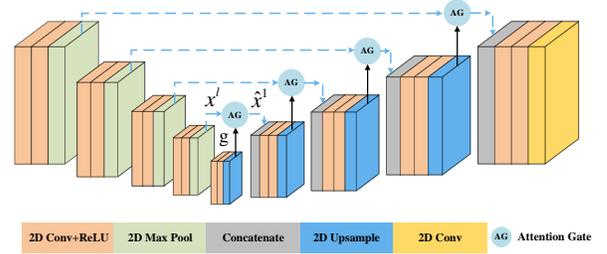


Figure 4: Architecture of our PGAU model.

Figure 4 illustrates the architecture of our PGAU model, a customized universal static IR drop analysis model, with corresponding layer details provided in the legend. The model consists of two sub-networks: (i) an encoder responsible for downsampling, and (ii) a decoder responsible for upsampling, interconnected by skip links. This architecture is rooted in a standard fully convolutional U-Net model, ensuring efficiency and independence of the input image size [6]. It demonstrates the ability to capture both local and global spatial neighborhood features. The 2D Convolutional layer and Rectified Linear Unit (ReLU) are employed to extract features from the input image. A 2D max-pooling layer is utilized to decrease the size of the feature map and reduce the number of parameters. Concatenation is employed to link two feature maps along the channel dimension. Additionally, a 2D upsample layer is used to enlarge the low-resolution feature maps from the encoder to match the dimensions of the original input image.

3.3.1 Encoder.

The encoder utilizes a series of 2D convolutional and max-pooling layer pairs to extract key features from a high-dimensional input feature set. Considering that the IR drop prediction usually contains complex spatial relationships and subtle features, the eigenvalues of each instance are discrete and unevenly distributed. To adapt more flexibly to IR drop changes in different regions and improve the sensitivity to nonlinear features, PGAU uses a $3 \times 3 \times 3$ filtered 2D convolutional layer combined with maxpool twice. With more parameters, downsampling helps to understand trends in the input image more clearly.

3.3.2 Decoder.

The decoder is responsible for retrieving the location information of the features lost during downsampling. This is very important for the IR drop prediction because the location and value of the special region such as the hotspots are decisive for the IR drop prediction. To focus on specific objects while ignoring unnecessary areas, we make use of the attention gate (AG) in this process. Each layer in the upsampling path includes an AG that filters the corresponding down-sampled features before concatenating them with the upsampled features, as shown in Figure 4.

The AG is set on the skip connection for each layer in the decoder of PGAU. The features from the corresponding encoder layers (i.e., $x^l \in \mathbb{R}^{F_l}$, where F_l corresponds to the number of feature-maps in layer l) and the result of the former decoder layer (a gating vector $g \in \mathbb{R}^{F_g}$ used for determining focus regions) are the input of the AG. The output of $\text{AG}(x^l)$ is concatenated with g_l before the next upsampling. AG is characterised by a set of parameters Θ_{att} containing: linear transformations W_x, W_g, ψ and bias terms b_ψ, b_g . Because

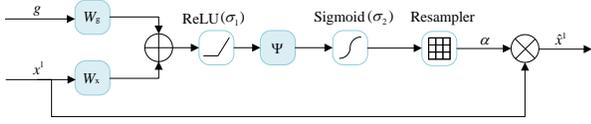


Figure 5: Illustration of the additive attention gate (AG).

soft-attention is differentiable, the weight W_x and W_g can be learned through the model network, like other parameters.

Figure 5 gives the structure of AG and can be formalized as:

$$q_{att}^l = \psi^T \left(\sigma_1 \left(W_x^T x^l + W_g^T g + b_g \right) \right) + b_\psi, \quad (5)$$

where g and x^l are transformed by W_g and W_x to gain prior knowledge, $\sigma_1(\cdot)$ is the ReLU activation function, and q_{att}^l is the attention weight used to calculate the attention coefficient. Acquiring prior knowledge in this manner enables the model to utilize the interest and attention gained during training, thereby improving the generalization. Subsequently, the results of the previous two operations are summed. Such an operation can enforce the same region of interest (ROI), that is, the hotspots in the IR drop distribution. Meanwhile, the respective different regions are also saved in the output as an aid. σ_1 is used to obtain attention weights to enhance the ROI. Then, we can get the attention coefficient α^l , which is defined as:

$$\alpha^l = \sigma_2 \left(q_{att}^l \left(x^l, g; \Theta_{att} \right) \right) \quad (6)$$

where $\sigma_2(\cdot)$ represents the sigmoid function, i.e., $\sigma_2(x_c) = \frac{1}{1+\exp(-x_c)}$. The role of σ_2 is to enable the model to dynamically adjust the attention to input features, which is used to improve the sensitivity to IR drop anomalies or hotspots. Finally, the attention coefficients α^l multiplied by x^l result in \hat{x}^l , a feature map containing prior IR drop information and ROI attention, represented as:

$$\hat{x}^l = \alpha^l x^l. \quad (7)$$

Borrowing from the work in [7], PGAU leverages a regression-like layer at the end of the decoder path, for IR drop prediction at 2D-spatial map level. A trained PGAU model is reusable, without retraining when facing new designs.

3.4 Loss Function with Label Distribution Smoothing

We perform a statistical analysis of the IR drop for a large number of PGs, and the distribution is shown in Figure 6. For a single PG, IR drops exhibit a skewed distribution with a long tail. For an amount of PG, most IR drops are concentrated in the lower part and a small number of hotspots are in regions with larger values. Overall, an unbalanced IR Drop distribution is shown.

We solve this problem using the label distribution smoothing method, which is based on the concept of re-weighting, particularly in addressing imbalanced classification problems [11]. For continuous regression labels, an empirical label distribution may not accurately represent a true label distribution because adjacent labels may not have clear boundaries. In situations with limited samples for specific labels, the model tends to prioritize learning features of labels with similar values [12]. Referring to the method in [12, 13], we calculate the empirical density distribution of labels in the training set. Then, we convolve it with a symmetric Gaussian kernel function to derive the smoothed effective label density distribution.

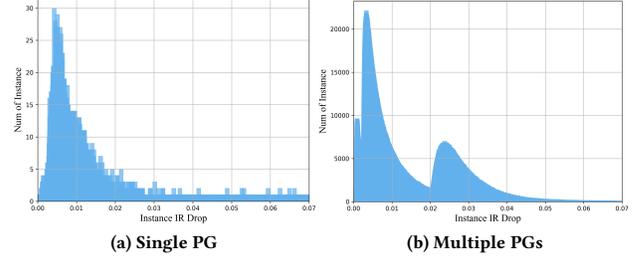


Figure 6: The imbalanced distribution of IR drop.

The reciprocal of the square root of this smoothed distribution serves as the weight for re-adjusting the loss function. The LDS calculation is:

$$\tilde{p}(y') = \int_y k(y, y') p(y) dy, \quad (8)$$

where y' is the value of the label, \hat{y} represents the predicted value, and $p(y)$ denotes the empirical density function of the label. We discretize the labels with a precision of 0.0001 and count the frequency of the value y to obtain $p(y)$. $\tilde{p}(y')$ represents the effective density function of y' , and the weight corresponding to a label with value y' is $\frac{1}{\sqrt{\tilde{p}(y')}} \cdot k(y, y')$, where the symmetric Gaussian kernel function

is defined as $k(y, y') \propto e^{-\frac{(y-y')^2}{2\sigma^2}}$. Considering the use of L1 loss, the re-weighted loss function is given by:

$$\text{loss}(\hat{y}, y') = \sum_{i=1}^n \frac{1}{\sqrt{\tilde{p}(y'_i)}} |\hat{y}_i - y'_i|. \quad (9)$$

4 EVALUATION

4.1 Experimental Settings

4.1.1 Baselines.

Published IR Drop Prediction Baselines. Our method is compared with ML-based IR drop models published in recent years, including XGBIR [3], PowerNet [5], IREDGe[6], and MAVIREC [7].

Competitive Image Segmentation Models. Our method is also compared with competitive models in the field of image semantic segmentation. U-Net [9] is a CNN model with an encoding-decoding structure, and its unique U-shape architecture can effectively capture multi-scale feature information. U-Net++ [14] further optimizes the performance of U-Net by introducing dense connections and multi-resolution paths. SETR [15] adopts a spatial-channel attention mechanism, which can effectively capture the global contextual information in the image. MobileNetV3 [16] is a lightweight CNN with an efficient model structure and low computational resource requirements. LR-ASPP [16] is a spatial pyramid pooling method used to efficiently process information at different scales in images.

4.1.2 Datasets.

Two datasets, CircuitNet [17] and ICCAD2023¹, are used for experimental evaluation. They are both datasets specialized for the static IR drop prediction task. The CircuitNet dataset is derived from real design, which is larger in scale and richer in information. Therefore, we pay more attention to the performance of the model on this dataset. To evaluate the performance of our model more broadly, we

¹<https://iccad-contest.org/>

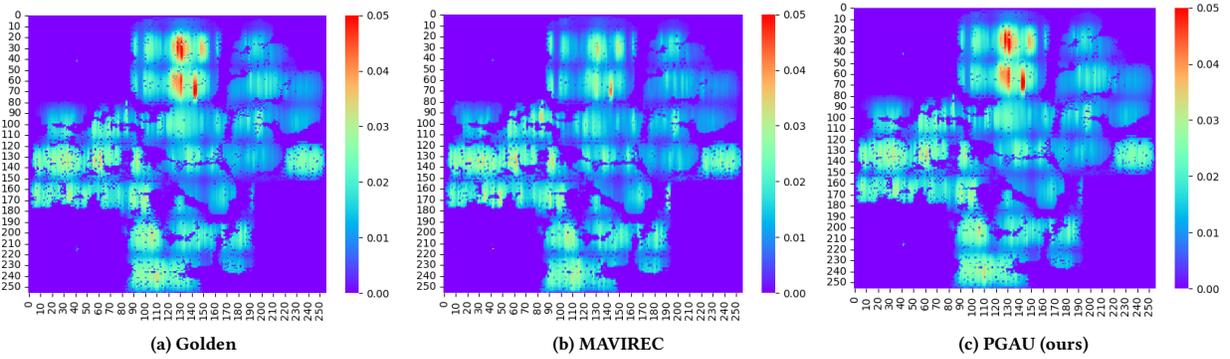


Figure 7: Visualization of IR Drop distribution of a PG.

also test it on the ICCAD2023 dataset. The CircuitNet dataset provides a new version for the Integrated Circuit EDA Elite Challenge², consisting of thousands of PGs generated from five industrial front-end designs at sub-10nm technology nodes, using different logic synthesis and physical design settings. It provides more complex information, including effective resistance data, power consumption data and other features, as well as static IR drop labels generated by GloryBolt³ (a commercial line-core EMIR analysis tool). The specific information is shown in Table 2. The IR drop distributions derived from the five types of industrial designs are relatively different, which places higher requirements on the generalization of the model. The ICCAD2023 dataset has 120 PG designs, including 100 fake designs and 20 real designs. The competition organizers use generative adversarial models [18] to generate fake yet close to realistic PGs. The dataset provides image-based data where each pixel in the image represents the current, effective distance, PDN density matrix, and IR drop of a PDN node in the lowermost metal layer in a $1\mu\text{m} \times 1\mu\text{m}$ region on the chip.

Table 2: Statistics of CircuitNet dataset

Front-end Design	RISCY-FPU	zero-risicy	RISCY	nvdla-small	Vortex-small
# PGs per design	2400	2400	2400	56	66
# Nodes per PG	>30000	>50000	>60000	>600000	>600000
Mean of IR drop	0.0028	0.0029	0.0034	0.0074	0.0076
Variance of IR drop	0.0003	0.0003	0.0004	0.0006	0.0006

4.1.3 Parameters and Metrics.

Our model is implemented within a PyTorch 2.0 framework on a 32-core CPU machine with 256GB RAM and one NVIDIA GTX3090 with 24GB RAM. All the baseline models are set at the same number of layers. Other hyperparameters, such as the optimizer or learning rate of the baseline, are carefully tuned for optimal performance.

As a measure of IR drop prediction accuracy, we use mean absolute error (MAE) and Pearson correlation coefficient (CC) to compare the model’s predictions with the ground truth. Since designers are more concerned about the maximum IR drop region of PG, its modeling error is extremely critical. Therefore, we also give the error where the IR drop is maximum, called MIRDE. Moreover, runtime is used to compare the efficiency of the models.

²<https://eda.icisc.cn/>

³https://www.phlexing.com/pro_services_desc.html?id=20

Table 3: Main results on CircuitNet and ICCAD2023 datasets

Method	CircuitNet dataset				ICCAD2023 dataset			
	MAE	CC	MIRDE	Runtime	MAE	CC	MIRDE	Runtime
U-Net [9]	0.0075	0.8704	0.0271	97s	0.3105	0.8413	0.8866	66s
U-Net++ [14]	0.0074	0.8736	0.0271	101s	0.3304	0.853	0.8195	71s
SETR [15]	0.0275	0.7239	0.0829	120s	1.1526	0.7694	2.2441	87s
MobileNetV3 [16]	0.0266	0.7252	0.0734	117s	1.0786	0.7712	2.3087	87s
LR-ASPP [16]	0.0258	0.7287	0.0717	117s	1.0025	0.7626	2.1358	85s
XGBIR [3]	0.0016	0.7816	0.0296	4s	0.6735	0.7841	1.2554	3s
PowerNet [5]	0.0087	0.8174	0.0308	140s	0.4214	0.7351	0.9671	79s
IREdGe [6]	0.0035	0.8153	0.0256	128s	0.3907	0.7690	0.8203	70s
MAVIREC [7]	0.0009	0.8854	0.0258	105s	0.3621	0.8651	0.8854	70s
PGAU (ours)	0.0007	0.9085	0.0231	97s	0.2995	0.8692	0.7691	68s

4.2 Main Results

To verify the superiority of our model, we conduct evaluation and comparison on two datasets respectively. First, the CircuitNet dataset is divided into training sets, validation sets, and testing sets with a ratio of 8:1:1, and the static IR drop prediction accuracy of each model is verified. The results are summarized in Table 3. Our method beats all baseline models and gains considerable advantages on each accuracy metric. As for the SOTA model (i.e., MAVIREC), the PGAU achieves better performance with an average metric improvement of 22.2% on MAE, 2.6% on CC and 10.5% on MIRDE. Figure 7 visualizes the predicted IR drop map given by our model and MAVIREC, comparing with the golden label. It can be intuitively seen that our model can provide more detailed predictions for the IR drop map with less error and trending closer to the label.

Due to the small number of PGs in the ICCAD2023 dataset, to ensure the reliability of the experiment, we perform 6-fold cross-validation, and the results are summarized in the right part of Table 3. The results show that our method achieves better performance, outperforming the SOTA model MAVIREC, with an average metric improvement of 17.3% on MAE, 0.5% on CC, and 9.7% on MIRDE.

4.3 Transfer Results

It can be seen from Table 2 that among the five industrial designs in the CircuitNet dataset, the first three are quite different from the last two, whether it is the number of PGs and nodes or the distribution of IR Drop. Therefore, in order to evaluate the transferability and generalizability of the model, the experiment uses the PGs generated from the first three designs as the training set and predicts the PGs generated from the last two designs. From the results in Table 4, our method is much better in generalization ability with better prediction

on the corresponding IR drop in the face of very different and never-seen PGs. This is because our model retains the key information of the PG data and the corresponding locations, while AGs focus on the key information and its vicinity. Referring to the AG already corrected by training, we can predict the trend and value of the IR drop in transferring work more accurately.

Table 4: Transfer Results on CircuitNet dataset

Method	MAE	CC	MIRDE	runtime
UNet [9]	0.0718	0.3525	0.2088	10s
UNet++ [14]	0.0676	0.3726	0.1849	12s
SETR [15]	0.5356	0.1962	0.2775	20s
MobileNetV3 [16]	0.3387	0.1731	0.3845	19s
LR-ASPP [16]	0.4673	0.2003	0.4803	18s
XGBIR [3]	0.0793	0.3277	0.3638	3s
PowerNet [5]	0.1996	0.2905	0.4472	20s
IREDDge [6]	0.0883	0.2914	0.3873	16s
MAVIREC [7]	0.0711	0.3374	0.2058	15s
PGAU (ours)	0.0647	0.3589	0.1689	12s

4.4 Ablation Study

To evaluate the effectiveness of each technique, a series of ablation experiments are performed, and the results are shown in Table 4.

Table 5: Ablation study results on CircuitNet dataset

Method	MAE	CC	MIRDE	Runtime
PGAU (w/o. $P_{composite}$)	0.0008	0.9048	0.0236	95s
PGAU (w/o. $P_{neighbour}$)	0.0010	0.8943	0.0244	95s
PGAU (w/o. $Count_{neighbour}$)	0.0010	0.8952	0.0239	95s
PGAU (w/o. LDS)	0.0007	0.9085	0.0231	97s
PGAU (w/o. AG)	0.0075	0.8704	0.0271	97s
PGAU (ours)	0.0007	0.9094	0.0223	97s

Selected Feature. First, we verify the effectiveness of our constructed features. Specifically, "w/o. feature's name" in Table 5 denotes removing this feature, and the corresponding accuracy shows a decrease. This experiment also validates the necessities of our feature importance analysis in Section 3.2.2.

Label distribution smoothing. The experiment examines the enhancement achieved through the label distribution smoothing in addressing imbalanced prediction tasks. The time loss caused by the LDS operation is negligible compared to the whole preprocessing and training process. Specifically, "w/o LDS" denotes training on the label without label distribution smoothing, revealing a marginal decline (i.e., Avg. CC=0.9% and Avg. MIRDE=3.5%).

Attention mechanism. It can be seen from the results in Table 5 that the model after removing all the AGs, that is, consistent with the U-Net model, decreases significantly in all measure metrics. This indicates that AG is very effective for our model.

5 CONCLUSION

In this work, we solve the problem of feature confusion and design a novel static IR drop prediction solution, verifying its superiority through extensive experiments. For future improvements, we suggest exploring more valuable composite features or pretraining a large vision model based on massive open-source datasets to reduce errors. Lastly, considering the fusion of tabular-based, image-based,

and graph-based ML models could further improve prediction performance.

ACKNOWLEDGMENTS

This work is supported in part by the Beijing Natural Science Foundation (No. 4244107), the National Key R&D Program of China (2022YFB2901100), and the National Natural Science Foundation of China (No.U20B2045, U1936220, 62192784, 62172052, 62002029, 61772082).

REFERENCES

- [1] Laung-Terng Wang, Yao-Wen Chang, and Kwang-Ting Tim Cheng. *Electronic Design Automation: Synthesis, Verification, and Test*. Morgan Kaufmann, 2009.
- [2] Haoxing Ren and Jiang Hu. *Machine Learning Applications in Electronic Design Automation*. Springer, 2022.
- [3] Chi-Hsien Pao, An-Yu Su, and Yu-Min Lee. XGBIR: An XGBoost-based IR Drop Predictor for Power Delivery Network. *IEEE/ACM Proceedings Design, Automation and Test in Europe (DATE)*, pp. 1307–1310, 2020.
- [4] Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System. *ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 785–794, 2016.
- [5] Zhiyao Xie, Haoxing Ren, Brucek Khailany, Ye Sheng, Santosh Santosh, Jiang Hu, and Yiran Chen. PowerNet: Transferable Dynamic IR Drop Estimation via Maximum Convolutional Neural Network. *IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC)*, pp. 13–18, 2020.
- [6] Vidya A Chhabria, Vipul Ahuja, Ashwath Prabhu, Nikhil Patil, Palkesh Jain, and Sachin S Sapatnekar. Thermal and IR Drop Analysis Using Convolutional Encoder-Decoder Networks *IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC)*, pp. 690–696, 2021.
- [7] Vidya A Chhabria, Yanqing Zhang, Haoxing Ren, Ben Keller, Brucek Khailany, and Sachin S Sapatnekar. MAVIREC: ML-Aided Vectorized IR-Drop Estimation and Classification. *IEEE/ACM Proceedings Design, Automation and Test in Europe (DATE)*, pp. 1825–1828, 2021.
- [8] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440, 2015.
- [9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241, 2015.
- [10] Nahian Siddique, Sidike Paheding, Colin P Elkin, and Vijay Devabhaktuni. U-Net and Its Variants for Medical Image Segmentation: A Review of Theory and Applications. *IEEE Access*, vol. 9, pp. 82031–82057, 2021.
- [11] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, vol. 106, pp. 249–259, 2018.
- [12] Yuzhe Yang, Kaiwen Zha, Yingcong Chen, Hao Wang, and Dina Katabi. Delving into Deep Imbalanced Regression. *International Conference on Machine Learning (ICML)*, pp. 11842–11851, 2021.
- [13] Jialv Zou, Xinggang Wang, Jiahao Guo, Wenyu Liu, Qian Zhang, and Chang Huang. Circuit as Set of Points. *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2023.
- [14] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. UNet++: A Nested U-Net Architecture for Medical Image Segmentation. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support (DLMIA)*, pp. 3–11, 2018.
- [15] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, et al. Rethinking Semantic Segmentation From a Sequence-to-Sequence Perspective With Transformers. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6881–6890, 2021.
- [16] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, et al. Searching for MobileNetV3. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1314–1324, 2019.
- [17] Zhuomin Chai, Yuxiang Zhao, Wei Liu, Yibo Lin, Runsheng Wang, and Ru Huang. CircuitNet: An Open-Source Dataset for Machine Learning in VLSI CAD Applications With Improved Domain-Specific Evaluation Metric and Learning Strategies. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 42, no. 12, pp. 5034–5047, 2023.
- [18] Vidya A Chhabria, Kishor Kunal, Masoud Zabih, and Sachin S Sapatnekar. BeGAN: Power Grid Benchmark Generation Using a Process-portable GAN-based Methodology. *IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, pp. 1–8, 2021.