DigWise Technology Corporation, LTD.

DTCO.VS DM-VS™ User Guide

Table of Contents

1. Introduction

The DM-VS™ employs a diffusion model as its core framework for wafer-level virtual data generation. Introduces the Denoising Diffusion Probabilistic Models (DDPM), which generate higher-quality silicon wafer data and overcome GAN limitations in multi-core chip performance features. Evaluating data distribution and quality with JS divergence and Fréchet Inception Distance (FID), the results show that the diffusion model accurately extracts feature distributions from silicon wafer data, generating numerous samples to support deeper analysis and accelerate the DTCO process. Compared to GAN, the diffusion model's generated wafers exhibit a data distribution closer to real data, with a JS divergence similarity of 0.987 and an FID of 6.28.

2. Denoising Diffusion Probabilistic Model (DDPM)

The Denoising Diffusion Probabilistic Model (DDPM) is a generative model that refines noisy samples into high-quality data by gradually removing noise. Widely successful in text and image generation, DDPM produces highly detailed and realistic results. Its core concept involves reversing the diffusion process, starting from noise samples and gradually correcting them to match real data distributions, yielding high-quality outputs. Fig. 1 illustrates the DDPM process, which includes both the forward and reverse processes, each involving T steps.

In the forward process, DDPM gradually adds Gaussian noise to the data, transforming it into a simpler distribution (usually Gaussian). This is done step by step, with the initial wafer sample $w0$ having noise added at each step, eventually turning into pure Gaussian noise wT. The network learns how to add noise, enabling accurate predictions in the reverse process.

In the reverse process, the model gradually reverses the forward process, recovering the original data distribution from the noisy samples. Specifically, the generation of the wafer begins with the noisy sample wT, which is then input into the neural network along with the time step t=T−1 to predict the noise added at the current step. Subsequently, the predicted noise is subtracted from wT to obtain wT−1. This process is repeated for T steps, eventually transforming the noisy sample wT into a high-quality wafer sample $w0$.

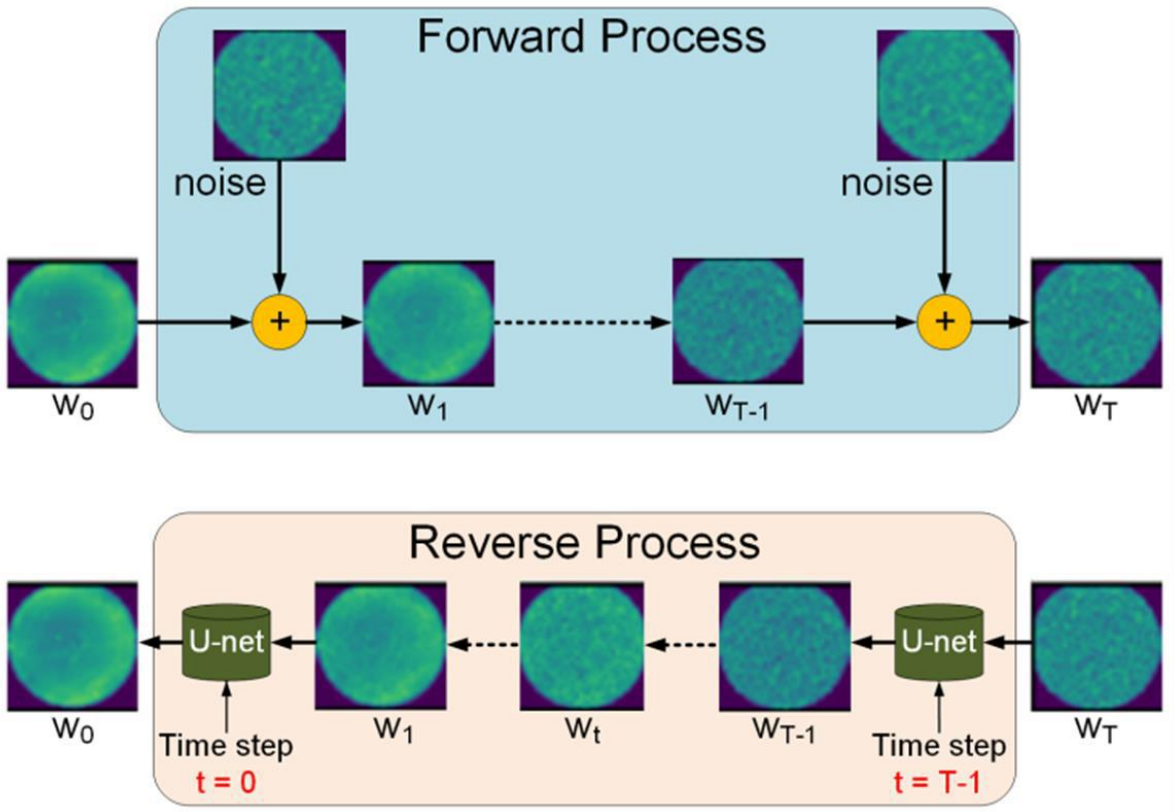

Fig. 1 Forward and Reverse Processes of the Diffusion Model

U-Net, commonly used for image segmentation and generation, has the structure shown in Fig. 2. The input passes through a convolutional layer to expand the channels to 16. The model contains two downsampling blocks that increase the channels to 64, followed by two upsampling blocks that restore the resolution while reducing the channels back to 16. A final convolutional layer generates the output with the desired dimensions. Fig. 3 provides further details of the ResNet block.

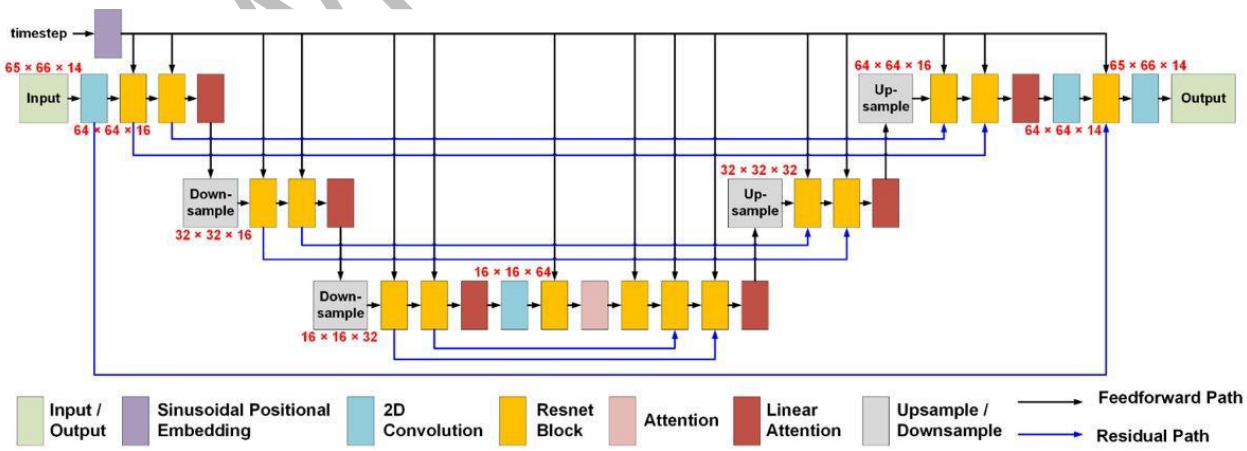

Fig. 2 U-Net of Diffusion Model

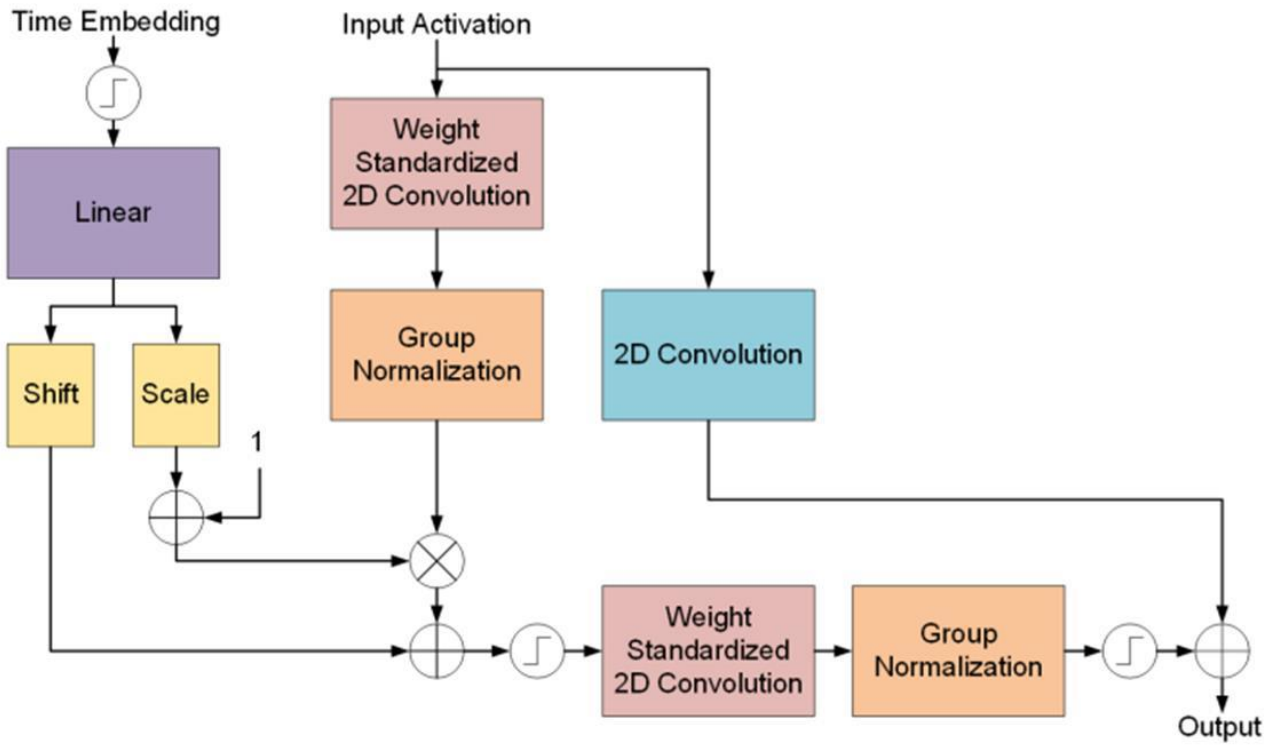

Fig. 3 Illustration of a ResNet Block

3. Diffusion Model Performance Evaluation

Introduces wafer-level evaluation metrics in addition to chip-level analysis. Unlike chip-level analysis, wafer-level focuses on chips from the same wafer, allowing for a more effective evaluation of whether the diffusion model successfully captures the complexity of the training data. This provides a more objective way to compare the data generated by the diffusion model with that generated by GAN.

In image generation tasks, FID is commonly used to evaluate the quality and diversity of generated data relative to real data. A lower FID indicates that the generated data more closely resembles the real data, reflecting higher generation quality. The FID is calculated by passing both real and generated data through a pre-trained Inception-v3 model and comparing their feature distributions. Fig. 4 illustrates the process of transforming wafer data into the dimensions required for FID calculation.

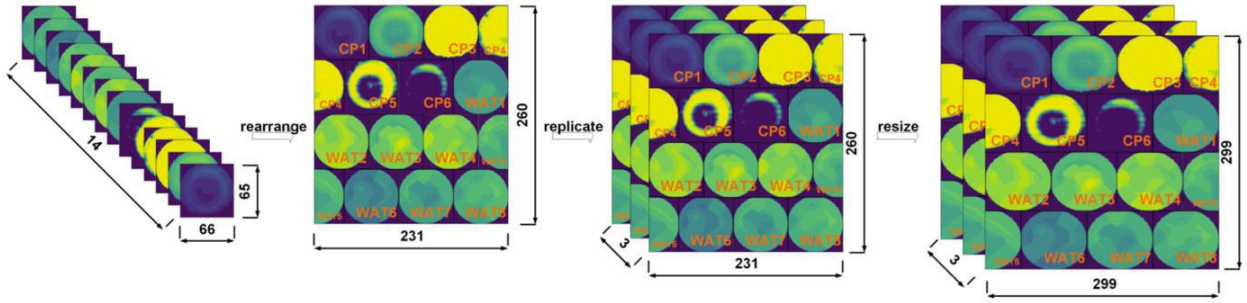

Fig. 4 Data Shape Transformation for FID Evaluation

3.1. Die-Level Analysis

At the chip level, scatter plots compare the joint distributions of multiple features between real and generated data. Fig. 5 shows the scatter plots for four feature pairs generated by the diffusion model, where the generated data points closely match the real data points, demonstrating high similarity in their joint distributions.

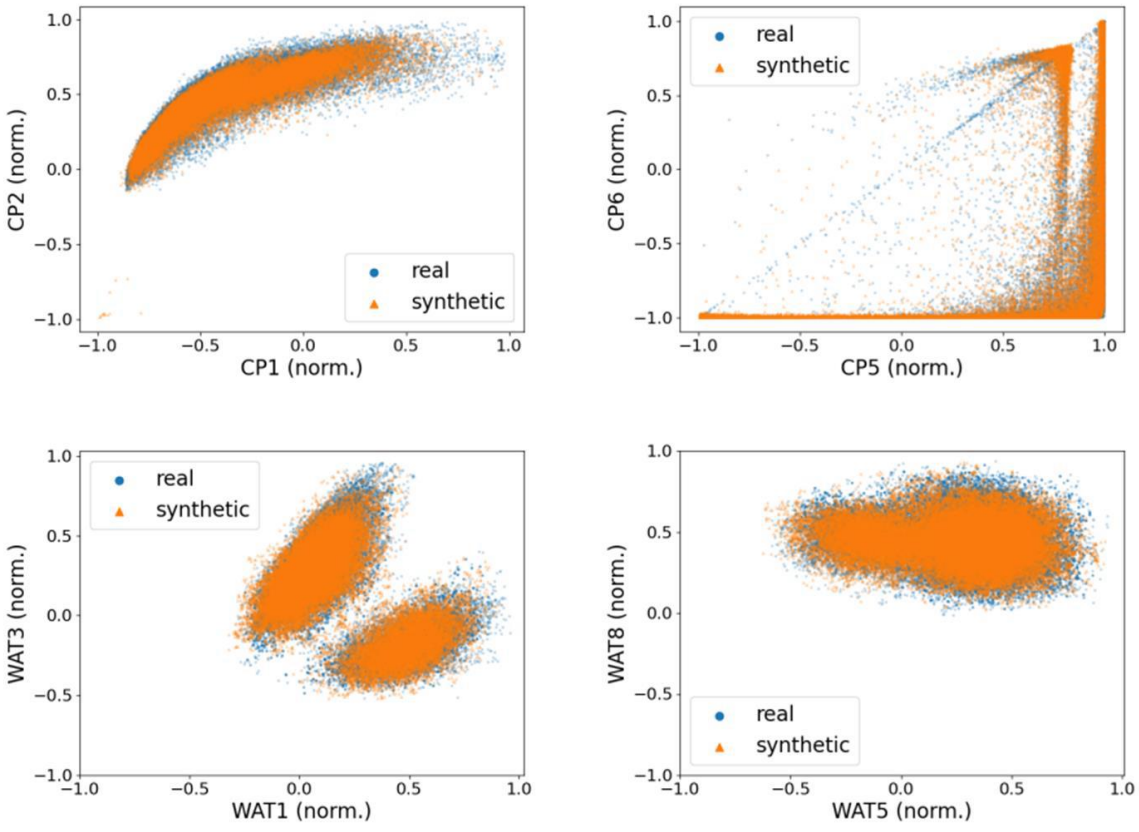

Fig. 5 Feature Scatter Plot for Diffusion Model Similarity

Fig. 6 shows the distribution of all feature values, with the JS divergence similarity indicated above each chart. In Fig. 6 (a), the GAN-generated distribution underperforms on features CP3, CP4, CP5, and CP6 due to the multiple peaks in their distributions. The mismatch in peak height and position leads to lower JS divergence similarity, and the GAN fails to capture the prominent peaks accurately.

Fig. 6 (b) shows the diffusion model's distribution, demonstrating its superiority in capturing the real distribution. For features CP3, CP4, CP5, and CP6, the diffusion model accurately reproduces the peak positions and heights. Specifically, for multi-core frequency performance features (CP3–CP6) with complex distributions like log or cosh, the GAN achieves a JS divergence similarity of 0.963, while the diffusion model improves this to 0.987, highlighting its superior performance on intricate distributions.

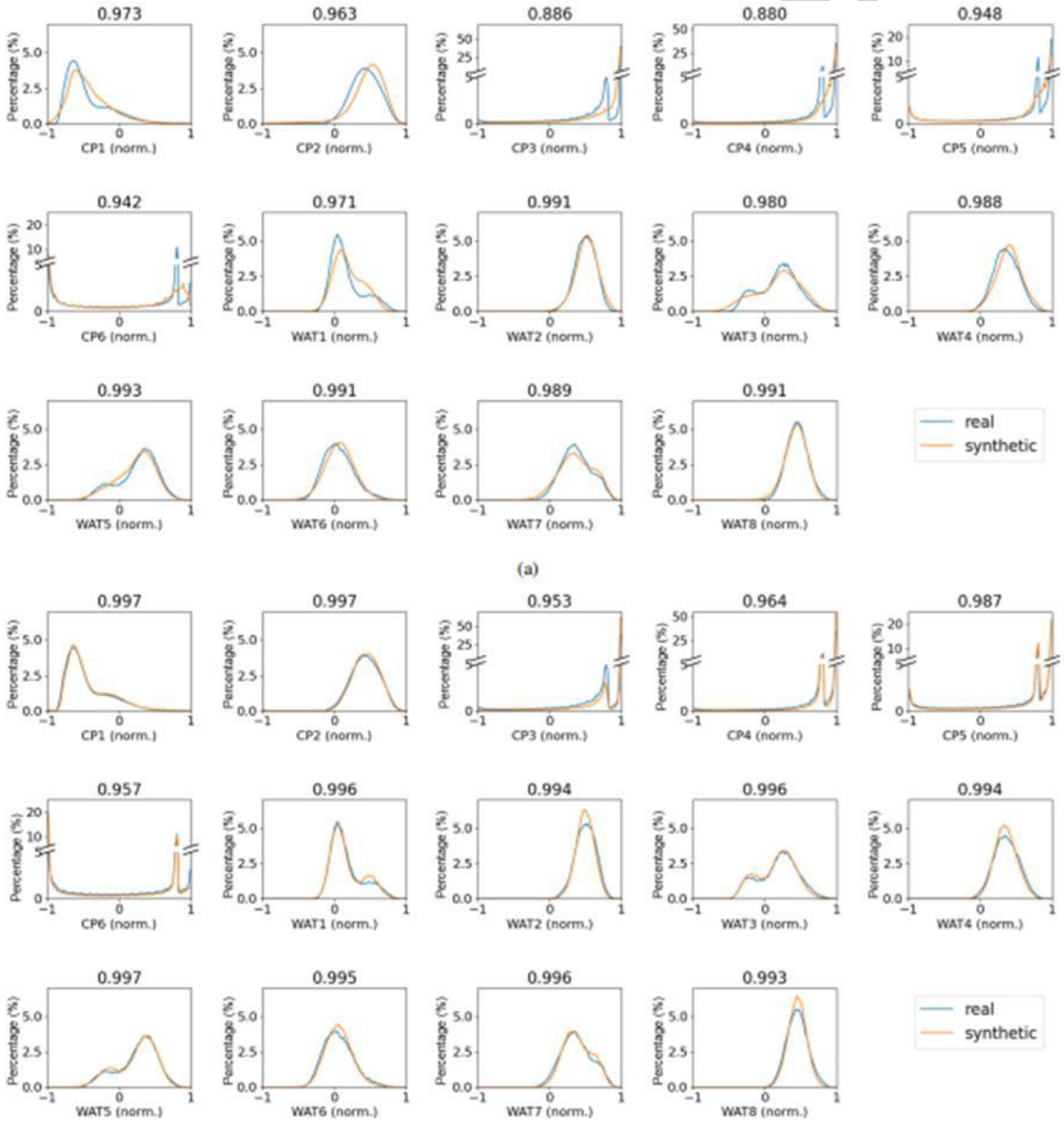

Fig. 6 Feature PDFs of GAN (a) and Diffusion Model (b)

3.2. Wafer-Level Analysis

At the wafer level, we assess the differences between adjacent chips along a specific direction to evaluate how well the spatial variations in the generated data match those in the real data. By calculating the average differences across wafers, we can quantify the similarity between real and generated data.

For simplicity, this analysis focuses on the secant along the horizontal direction of the wafer, examining the average differences between adjacent chips in that direction. Fig. 7 shows the average difference analysis of CP1 and WAT1 generated by the GAN and diffusion model. The x-axis represents the horizontal coordinates, and the y-axis shows the average difference. The light blue area indicates one standard deviation range of the real data's average difference, providing an intuitive comparison between generated and real data.

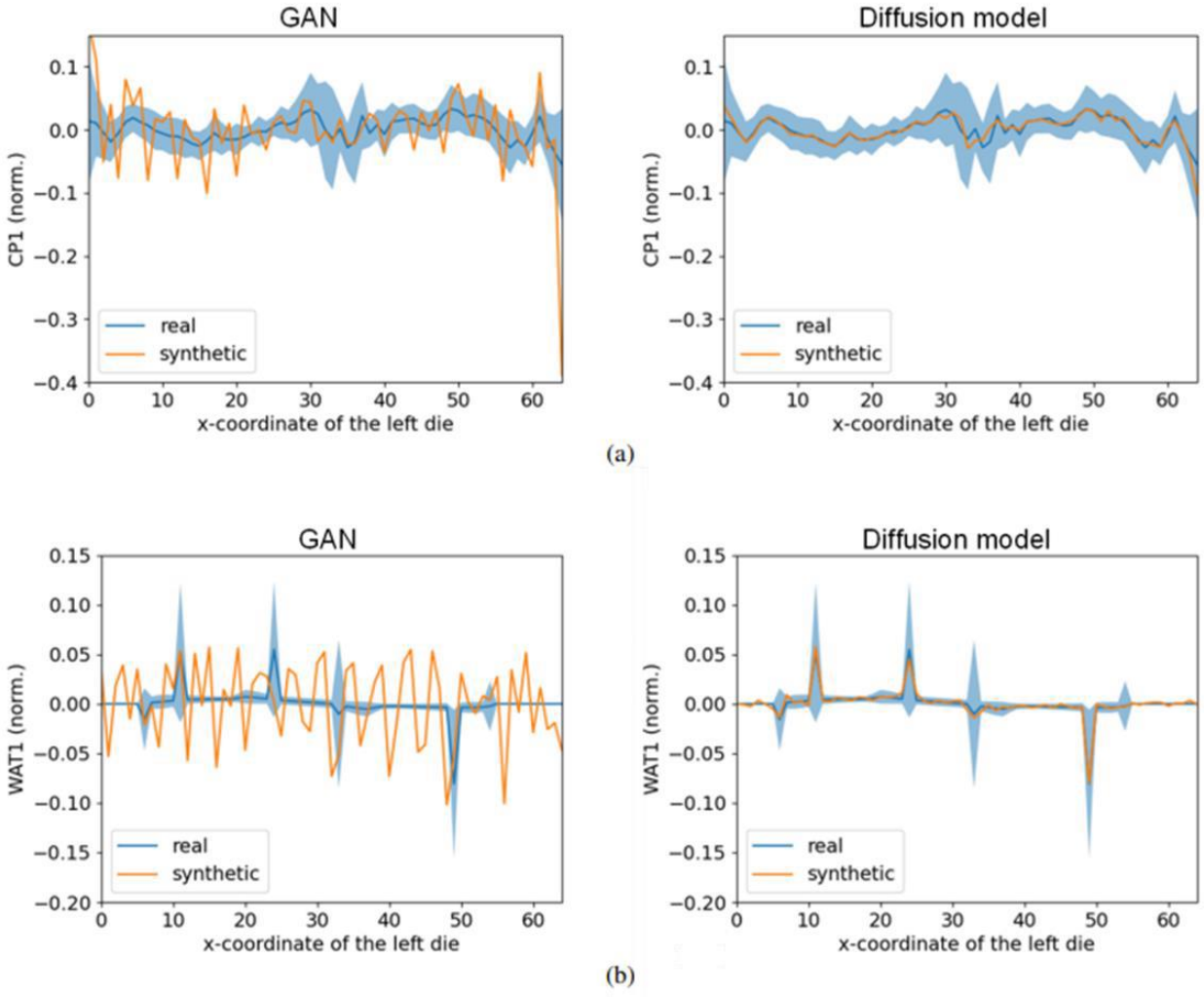

Fig. 7 Feature PDFs of GAN (a) and Diffusion Model (b)

Fig. 7 (a) shows the average difference of CP1. For GAN-based methods, the generated data's average difference often falls outside the standard deviation range, with significant fluctuations indicating inaccurate patterns. In contrast, the diffusion model keeps the average difference consistently within the standard deviation range. Fig. 7 (b) shows similar results for WAT1, where GAN-generated data exhibits notable fluctuations, while the diffusion model captures the peak differences of real WAT1 data. The average difference from the diffusion model stays within one standard deviation of real data, showing similar spatial variations in both horizontal and vertical directions.

Table I compares the FID (Frechet Inception Distance) of virtual wafers generated by GAN and the diffusion model to real data. The real data's FID is 1.39, calculated by splitting the wafer data into two equal parts. GAN's FID is 55.13, indicating difficulty in capturing multi-modal distributions, while the diffusion model's FID is 6.28, closely matching real data and showing a significant improvement in generation quality. Overall, while GAN performs well on some features, it struggles with multi-modal distributions. The diffusion model accurately simulates real data's multi-feature distribution, with high consistency in chip distribution and horizontal secant differences.

TABLE I: Quality Comparison of Generated Data

Metric	GAN	Diffusion model
Average JS divergence similarity	0.963	**0.987**
FID	55.13	**6.28**

4. Virtual Silicon Data Format

To ensure seamless integration and analysis within the DM-VS™, it is essential to understand the required data format for Chip Probing (CP) and Wafer Acceptance Test (WAT) data. The following sections outline the expected structures for data type:

- **File Type**: ZIP and CSV.
- **Required Columns**:
  - **LWID**: A distinctive identifier assigned to each generated wafer for tracking and analysis. (Lot Id. + Wafer No.)
  - **X**: Represents the horizontal coordinate, uniquely identifying the chip's position on the generated wafer.
  - **Y**: Represents the vertical coordinate, uniquely identifying the chip's position on the generated wafer.
  - **Features**: Measurement parameters (refer to TABLE II and Fig. 8).

TABLE II: Generated virtual silicon features in the dataset.

Feature	Description	Unit
CP1	Leakage current	μA
CP2	Chip speed	Hz
CP3	Functional accuracy at 300MHz	%
CP4	Functional accuracy at 400MHz	%
CP5	Functional accuracy at 500MHz	%
CP6	Functional accuracy at 600MHz	%
WAT1	Gate threshold voltage of the low threshold NMOS	V
WAT2	Gate threshold voltage of the low threshold PMOS	V
WAT3	Gate threshold voltage of the ultra-low threshold NMOS	V
WAT4	Gate threshold voltage of the ultra-low threshold PMOS	V
WAT5	Drain current of the low threshold NMOS	mA
WAT6	Drain current of the low threshold PMOS	mA
WAT7	Drain current of the ultra-low threshold NMOS	mA
WAT8	Drain current of the ultra-low threshold PMOS	mA

LWID	X	Y	CP1	CP2	CP3	CP4	CP5	CP6	WAT1	WAT2	WAT3	WAT4	WAT5	WAT6	WAT7	WAT8
genData-1	3	27	3.741	2699.161	415.642	454.174	464.529	430.347	0.127	0.123	0.048	0.075	4.448	4.027	10.775	7.455
genData-1	3	28	3.69	2769.055	414.792	455.175	463.871	423.284	0.126	0.123	0.047	0.075	4.445	4.033	10.847	7.465
genData-1	3	29	3.899	2837.581	396.38	448.394	457.458	424.834	0.126	0.124	0.048	0.075	4.45	3.964	10.82	7.496
genData-1	3	30	3.998	2848.2	422.024	455.271	460.276	397.482	0.127	0.124	0.047	0.075	4.395	3.967	10.84	7.506
genData-1	3	31	3.474	2704.473	445.977	458.816	462.295	413.233	0.127	0.124	0.047	0.076	4.373	3.958	10.818	7.518
genData-1	3	32	3.307	2701.763	452.429	461.856	460.52	329.241	0.127	0.124	0.049	0.076	4.353	3.949	10.743	7.519
genData-1	3	33	3.312	2678.831	450.663	464.097	451.603	215.344	0.127	0.123	0.048	0.076	4.387	3.978	10.783	7.559
genData-1	3	34	3.282	2677.911	448.199	462.331	456.582	277.068	0.127	0.123	0.047	0.076	4.366	3.957	10.792	7.56
genData-1	3	35	3.124	2629.331	447.324	461.937	449.28	195.471	0.127	0.123	0.048	0.076	4.393	3.957	10.728	7.526
genData-1	3	36	2.978	2612.604	453.462	460.84	437.85	147.733	0.128	0.122	0.047	0.076	4.377	3.958	10.741	7.539
genData-1	3	37	2.957	2579.561	449.839	459.419	442.787	127.626	0.127	0.121	0.047	0.076	4.362	3.959	10.701	7.554
genData-1	3	38	3.054	2625.559	447.56	463.512	452.681	124.029	0.128	0.121	0.048	0.076	4.358	3.946	10.673	7.505
genData-1	3	39	4.266	2590.8	442.496	455.201	452.296	191.388	0.127	0.121	0.047	0.075	4.389	3.974	10.748	7.511
genData-1	3	40	3.221	2511.235	442.528	459.99	457.661	121.853	0.128	0.121	0.047	0.075	4.366	3.956	10.768	7.497
genData-1	3	41	3.192	2575.551	443.485	453.556	460.591	214.117	0.129	0.122	0.047	0.076	4.337	3.922	10.792	7.481
genData-1	3	42	3.341	2540.776	402.23	424.933	444.301	300.042	0.129	0.12	0.047	0.076	4.336	4.01	10.854	7.482
genData-1	3	43	3.579	2515.6	443.838	456.725	465.544	363.565	0.129	0.121	0.048	0.075	4.351	3.949	10.8	7.484
genData-1	3	44	3.804	2548.631	443.743	459.015	467.971	310.41	0.129	0.122	0.048	0.076	4.398	3.974	10.82	7.504
genData-1	3	45	3.402	2348.923	440.055	458.225	463.798	264.295	0.128	0.124	0.047	0.076	4.431	3.911	10.906	7.513
genData-1	3	46	3.176	2228.373	384.894	393.103	352.435	130.737	0.125	0.125	0.048	0.075	4.724	3.963	10.854	7.706
genData-1	4	17	3.509	2146.707	170.892	215.822	210.77	96.017	0.126	0.123	0.049	0.075	4.661	3.903	10.741	7.664
genData-1	4	18	4.276	2403.186	421.754	451.038	464.898	360.905	0.128	0.125	0.049	0.075	4.586	3.96	10.887	7.629
genData-1	4	19	3.487	2566.804	437.044	458.959	461.165	250.047	0.129	0.12	0.048	0.076	4.427	3.971	10.782	7.577
genData-1	4	20	3.347	2521.76	402.128	405.949	410.823	225.052	0.13	0.12	0.048	0.076	4.375	3.98	10.841	7.464
genData-1	4	21	3.13	2555.888	438.773	459.238	457.02	238.413	0.129	0.122	0.048	0.075	4.386	3.961	10.787	7.528
genData-1	4	22	3.227	2664.797	437.033	457.475	462.982	293.982	0.128	0.122	0.048	0.075	4.39	3.987	10.844	7.495
genData-1	4	23	3.478	2718.018	442.095	460.662	463.368	342.826	0.128	0.123	0.048	0.075	4.424	3.979	10.849	7.473

Fig. 8 CP + WAT Virtual Silicon Data Format.

5. Virtual Silicon Data Visualization

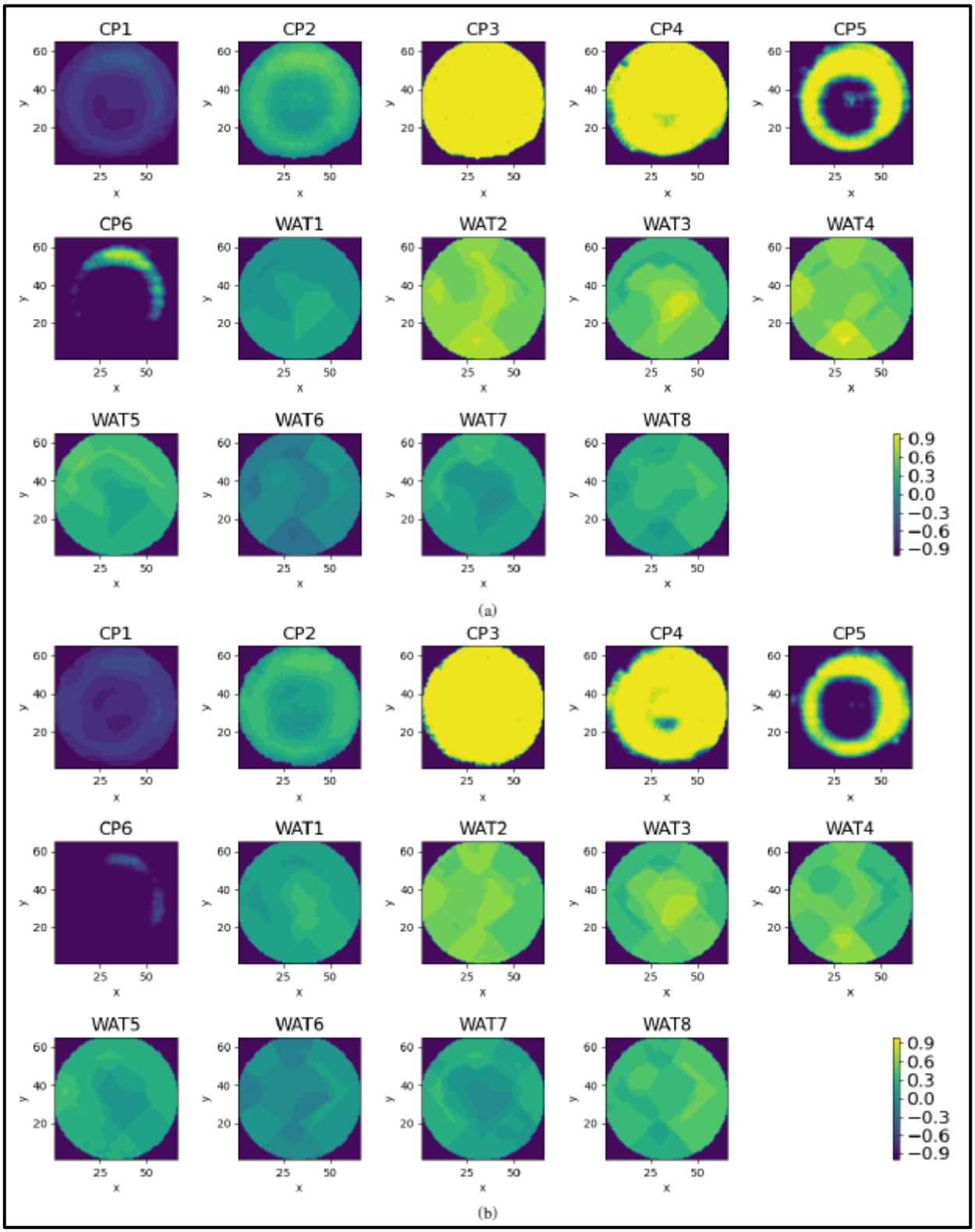

Fig. 9 Demonstrations of (a) a wafer in the training data, and (b) a wafer generated by the diffusion model.

6. Technical Insights

6.1. Describe the WAT Data Source

ANS>

In practice, unless the chip design company embeds a WAT test-key within the chip itself, only about 80 discrete test points (full-map) can be obtained during the pilot stage before mass production. After entering the mass production phase, the number of test data points provided by the foundry may significantly decrease, typically ranging from 9 to 13 points, and in some mature processes, it may be reduced to just 3 points. For example, in a 16nm-like process (see Figure 10), high-resolution CP SIDD data reveals that even within the same shot (5×7 dies), there can be significant differences in the electrical characteristics of the chips. Therefore, insufficient WAT sampling may lead to biased conclusions, which in turn could affect decision-making accuracy.

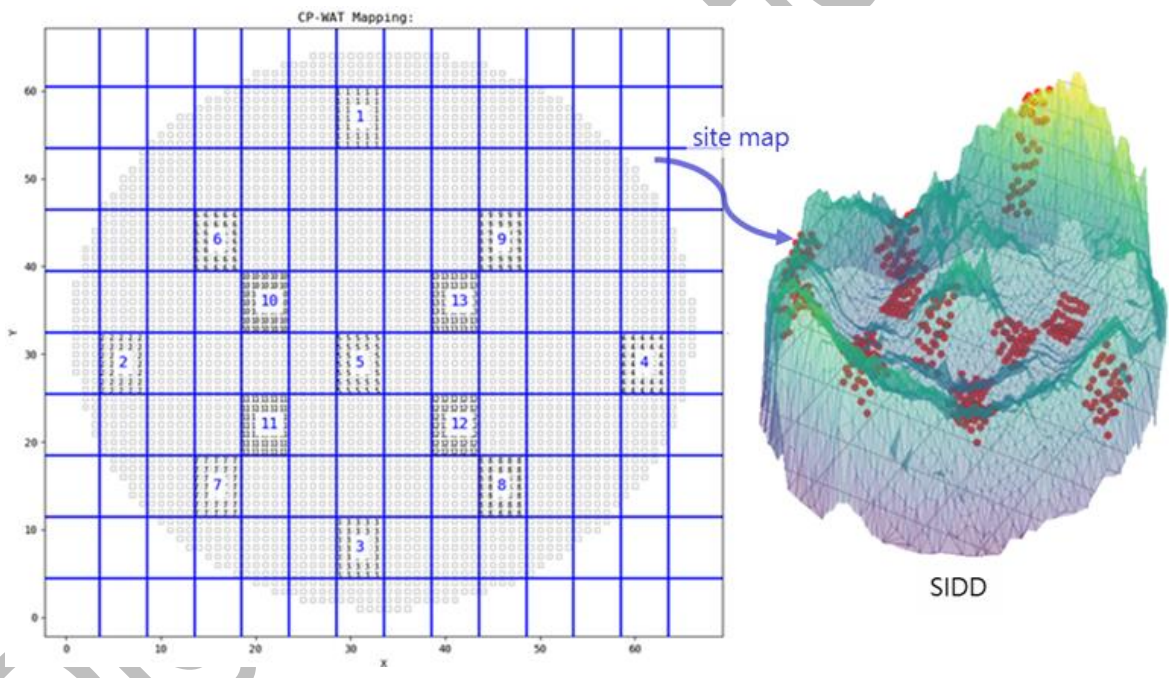

Fig. 10 Impact of Insufficient Data Sampling on Biased Generalization

To effectively enhance decision-making confidence, we assume that the physical electrical characteristics near the test points are similar. Therefore, data augmentation can be performed using methods such as nearest-neighbor similarity or linear regression models, for example, expanding 13 points to over 80 points (full-map), as shown in Figure 11.

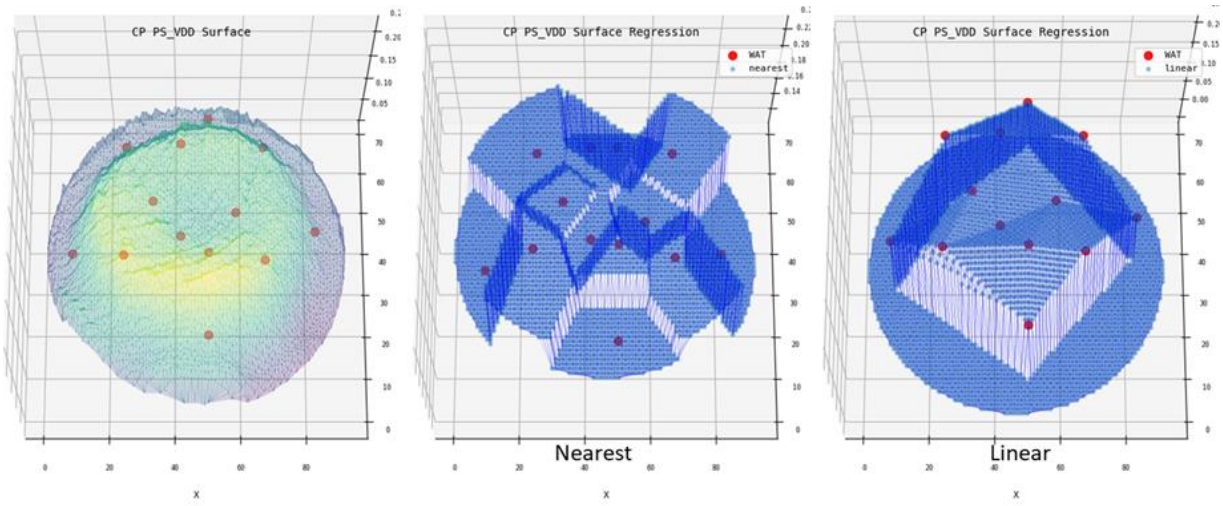

Fig. 11 Traditional WAT Feature Augmentation (Nearest and Linear Regression)

In another case of a 6nm-like process for a CPU chip, data augmentation using a linear regression model can increase the data resolution by approximately 40 times, thereby enhancing confidence in dynamic data correlation tracking and parameter tuning. However, the linear regression model tends to generate false distributions, as seen in the orange circle on the right side of Figure 12.

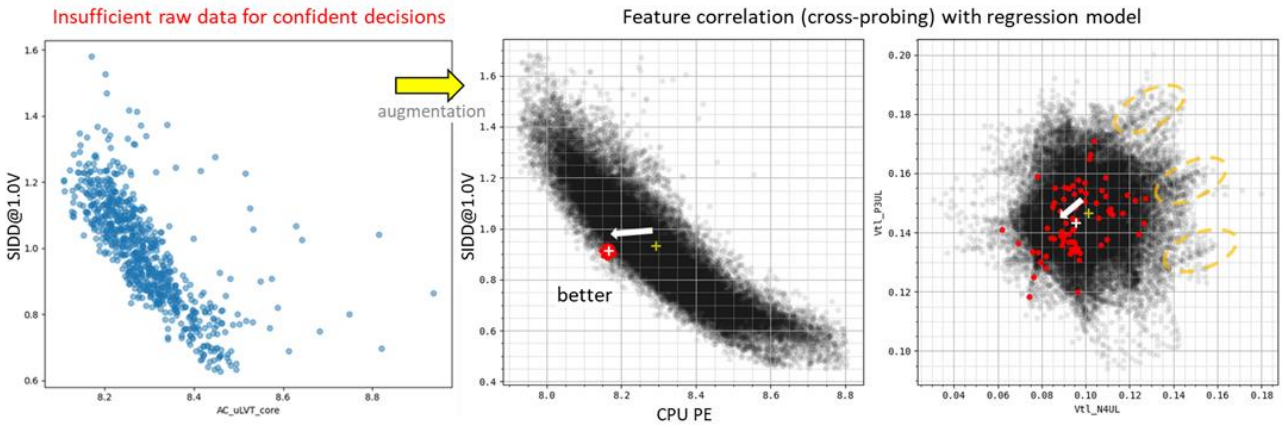

Fig. 12 Feature Augmentation with Regression Model

The above issues can be addressed through MDV (Multi-Valued Dependency) or GMM (Gaussian Mixture Model). However, traditional modeling methods struggle to capture the large, non-random structural characteristics of wafer-level uniformity. As shown in Figure 13, in an example of a 16nm-like process, other CP characteristics (such as Power-Short and RO) also exhibit distinct spatial structural features. These system-level non-random uniformity structures are difficult to accurately capture with only a few discrete test points.

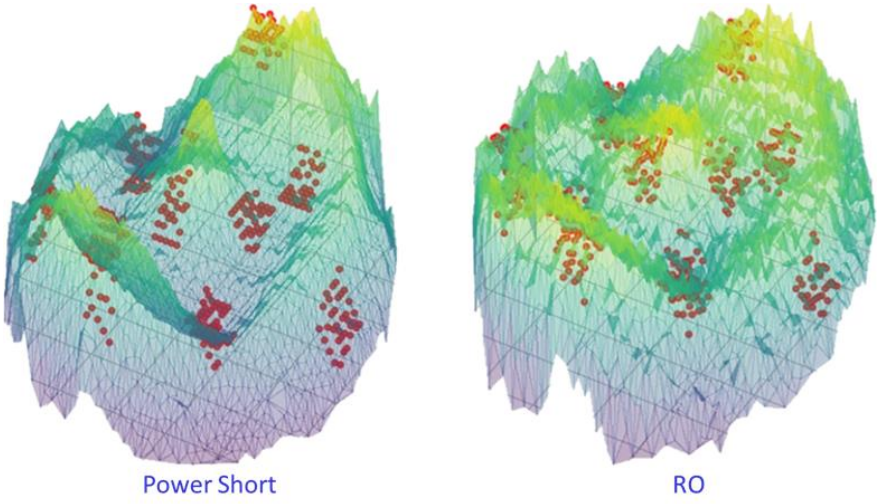

Fig. 13 CP Feature Uniformity

As the number of WAT sampling points increases, the wafer-level uniformity structure becomes more clearly defined. For example, in another 16nm-like process case shown in Figure 14, test data from 80 points (full-map) gradually reveals the uniformity characteristics within the wafer.

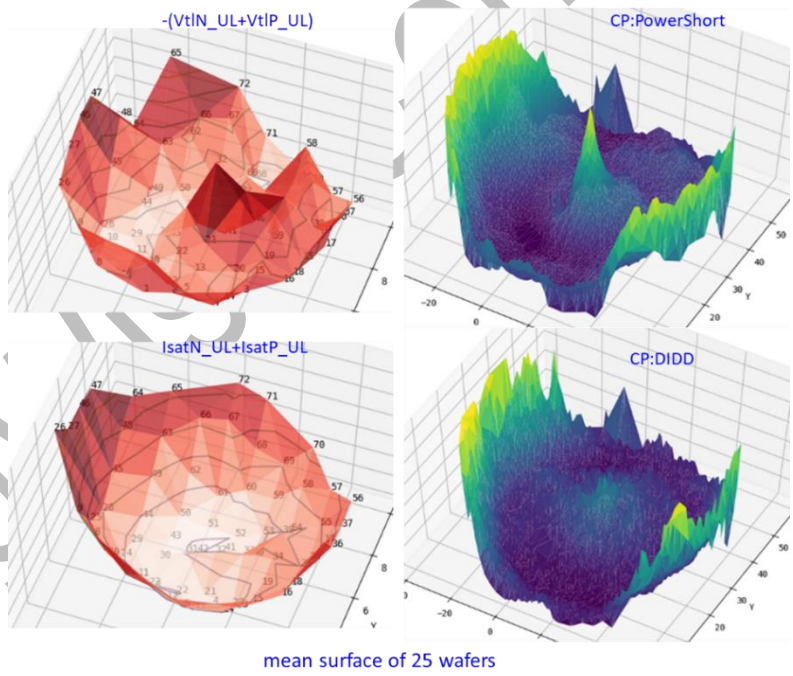

Fig. 14 WAT & CP Uniformity Correlation

Improving WAT resolution is a topic worth in-depth research and optimization. As shown in Figure 15, the data augmentation method used in this study combines cubic regression and nearest-neighbor averaging to approximate the nonlinear structure of the wafer. The goal is to construct a network using a Diffusion-Model to generate near-realistic multidimensional

data while ensuring that the relationships between features remain highly consistent with the original data, rather than directly addressing the issues of ultra-high WAT resolution or DTCO (Design-Technology Co-optimization).

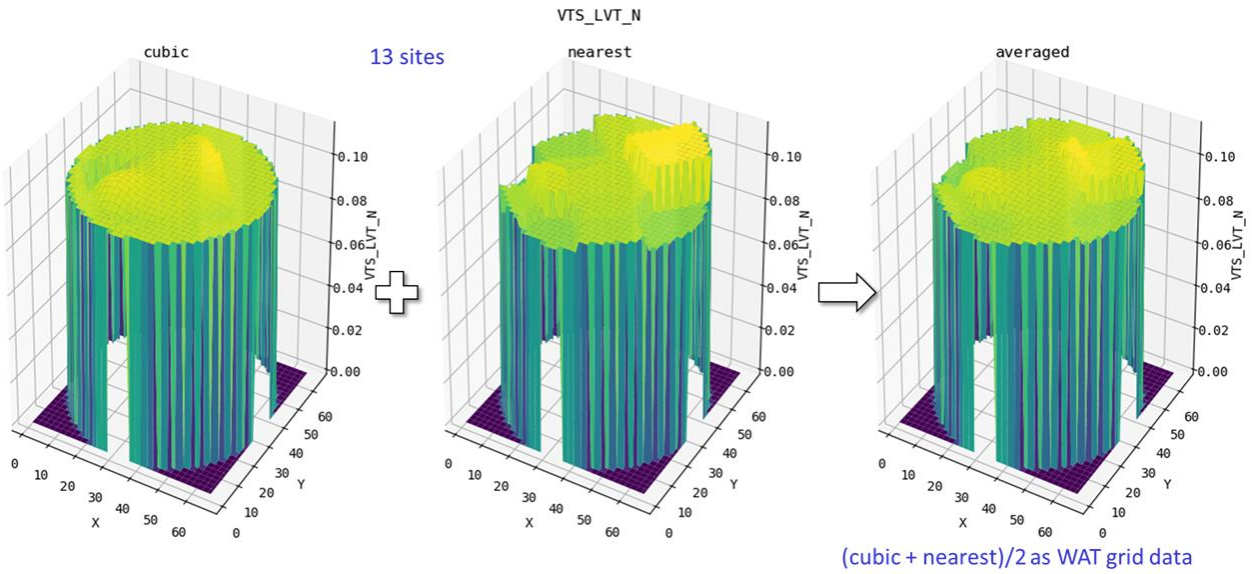

Fig. 15 WAT Feature Augmentation in the Training Set

Although these methods can visualize decision boundaries and enhance decision-making confidence, traditional regression methods often produce false distributions when modeling the nonlinear uniformity relationships at the wafer level. Therefore, accurately capturing the wafer-level uniformity structure and effectively improving WAT resolution is another critical area of focus. Our team is currently actively exploring promising solutions, but these will not be further discussed in this documentation.

6.2. Why Choose LVT and ULVT Transistors for Low Power?

ANS>

The example used in this documentation is a high-speed computing chip (BTC) operating at the near-threshold voltage, with its core being ULVT. However, the focus of this study is not on the chip's characteristics themselves, but rather on how to observe the relationships between WAT and CP characteristic features in a multidimensional space through spatial coordinate alignment. For example, in some foundries, the impact of PMOS devices on chip leakage current is significant, prompting corresponding adjustments. As for which feature parameters should be selected for parameter tuning, voltage compensation, mass production strategies (binning), OCV evaluation, SPICE-Silicon correlation analysis, and

design sign-off methodology optimization, these are topics worthy of further research.

The multidimensional distribution of real chip data is not a single Gaussian distribution but takes various forms of probability distributions, such as skew-normal, log-normal, log, catenary, and even combinations that include hyperbolic cosine (cosh) distributions. Furthermore, the relationships between high-dimensional parameters are not purely linear and cannot be perfectly modeled by GMM (Gaussian Mixture Model). Traditional data modeling methods often fail to capture the wafer-level uniformity structure, and this non-randomness makes traditional design methods unreliable and overly pessimistic.

This documentation aims to explore how to effectively solve the above issues using a simple GAN, while capturing the distributions, relationships, and uniformity structure in high-dimensional space, making the generated virtual data more realistic and natural. However, GANs face limitations when handling non-Gaussian distributions such as double-logarithmic or hyperbolic cosine. The Diffusion-Model, on the other hand, can significantly improve this issue, further supporting the generation of higher-dimensional data and ensuring data quality. This method is not only applicable to the 16-dimensional data discussed in this documentation (including x, y), but can also be extended to more feature dimensions to enhance the overall modeling accuracy and applicability.

6.3. State the VT/Id Being Applied

ANS>

The availability of WAT data varies by foundry. The features used in this documentation are defined as Vt saturation and Id saturation based on the foundry in this example. However, different foundries may have varying measurement methods and definitions. The purpose of this study is solely to demonstrate the application of the method and is not limited to the use of specific features.

7. Getting Started

7.1. Beginner Users

Access: Free download of one suitable sample from the website.

Purpose: To explore and familiarize themselves with the system or product.

7.2. Advanced Users

Access: Option to download 5 or 10 samples based on the chosen subscription plan.

Purpose: To gain deeper insights or leverage additional resources for professional use.

7.3. Custom Users

Access: Users requiring more samples are encouraged to contact us directly for customized solutions.

8. Customer Support and Assistance

For further assistance or to report any issues you may encounter, please reach out to our dedicated support team. Our team is committed to providing timely solutions and ensuring your experience with our system is seamless.

Contact Information:

- **AnswerXpert QA Forum:** http://172.17.20.61/post_message4.php
- **Operating Hours:** Monday to Friday, 9:00 AM - 6:00 PM (GMT)

Feel free to contact us with any questions, feedback, or concerns. We value your input and are here to help you resolve any challenges effectively.