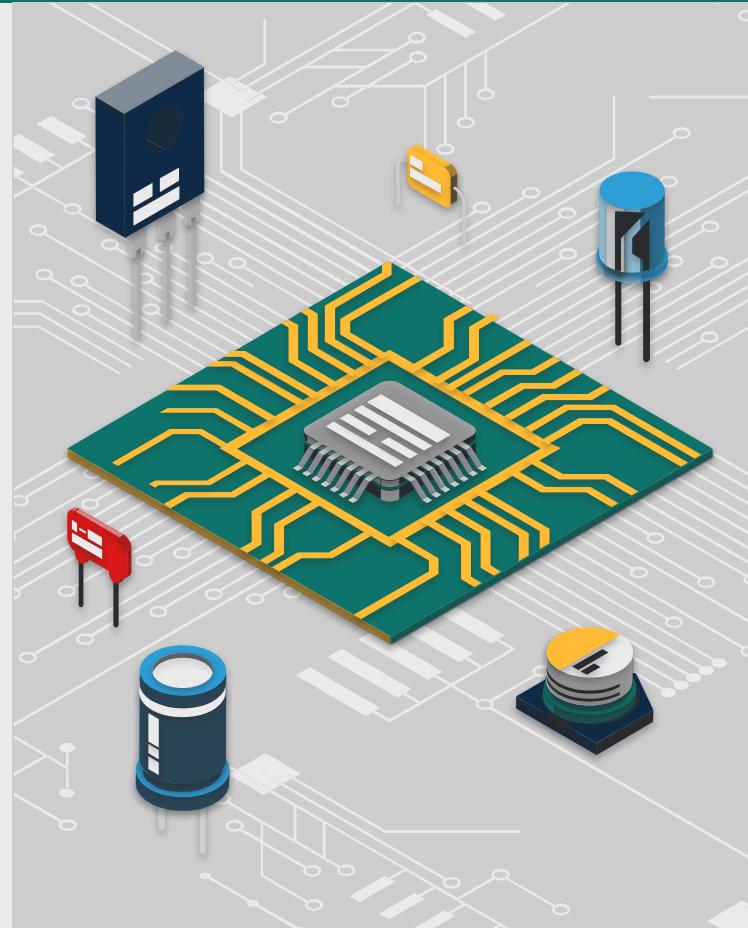


# Factors Influencing Lithography Mask Optimization

13기 백승이  
14기 권나연  
14기 여준호  
15기 배연욱  
15기 백가은



Semiconductor

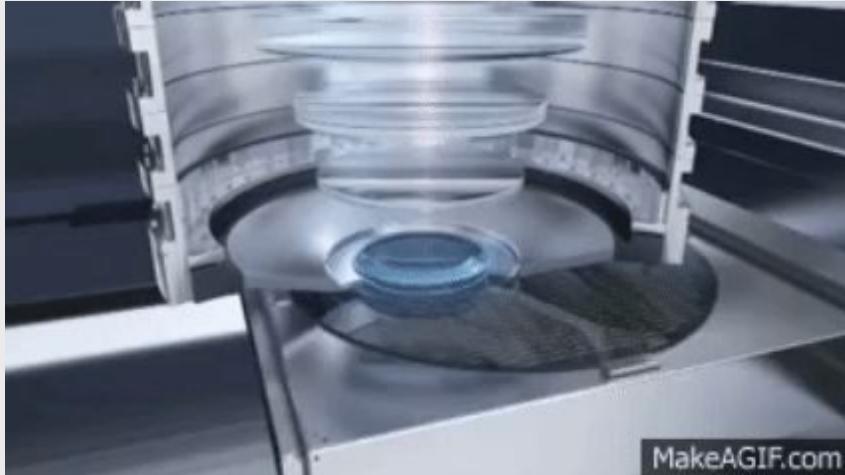
# Table of contents

- 01      Background**
- 02      Data Feature Analysis  
: StdMetal VS StdConact**
- 03      Price Perspective  
: Via VS Metal**
- 04      FFT-Based Efficient Viewing Method**

01

# Background

# Semiconductor



MakeAGIF.com



# Semiconductor Manufacturing process

전공정(Front-end of line)



# Lithography

빛을 이용하여 실리콘 웨이퍼 위에 복잡한 미세 회로 패턴을 형성하는 공정

## STEP1. 감광액 도포(Coating)

웨이퍼 표면에 감광액(빛에 반응하는 화학 물질)을 얹고 균일하게 도포

## STEP2. 노광(Exposure)

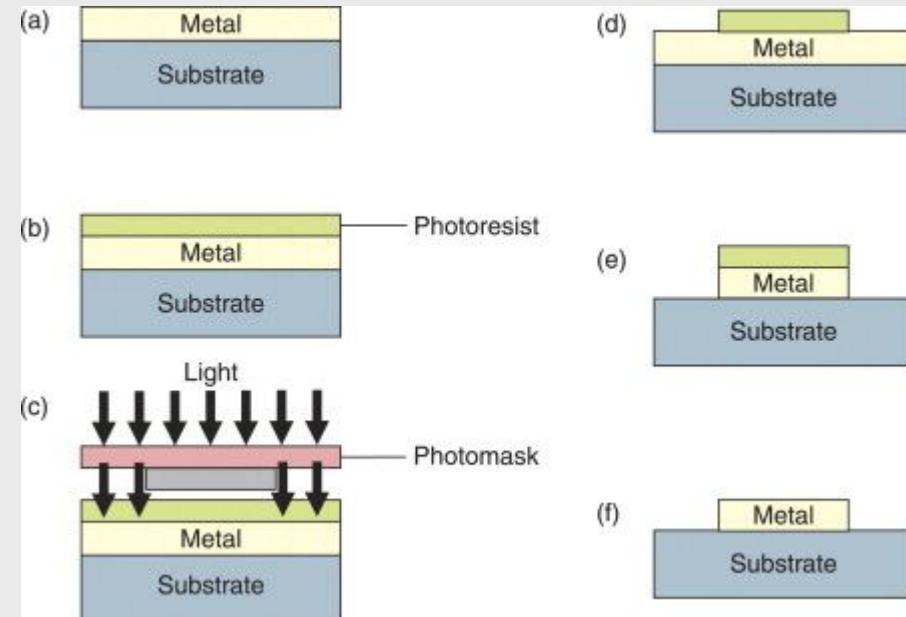
회로 패턴이 담긴 마스크에 빛을 투과시켜 웨이퍼에 전사

## STEP3. 현상(Development)

화학 용액을 이용해 빛을 받은  
or 받지 않은 부분의 감광액을 제거, 패턴 형상화

## STEP4. 식각(Etching)

감광액이 제거된 부위의 실리콘을  
깎아내어 실제 물리적 회로 형성



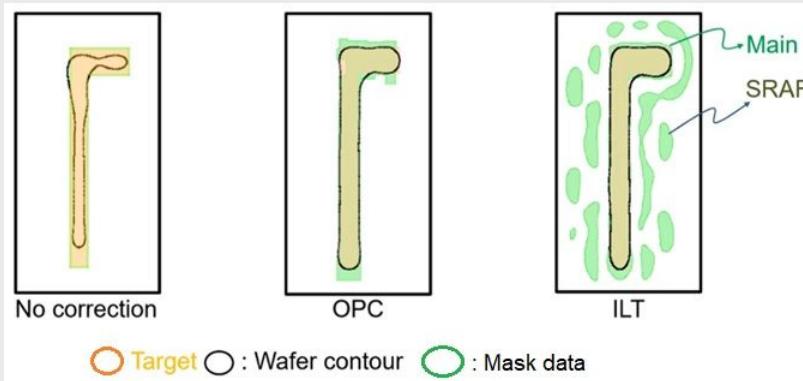
# Inverse Lithography Technology

## 공정 미세화의 한계

반도체 공정 미세화로 회로 선폭(CD)이 광원 파장보다 작아짐 → 기존 광학 리소그래피의 물리적 한계 도달

## 발생 현상

빛의 회절(diffraction)·간섭(interference) 현상 증가  
→ 웨이퍼 상에서는 번짐·왜곡된 형태로 전사됨



## → Inverse Lithography Tech(ILT)

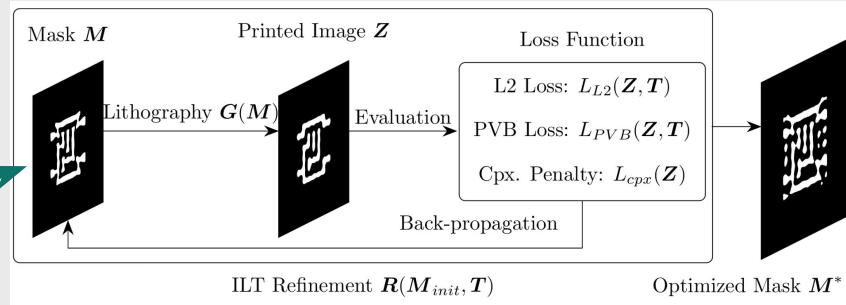
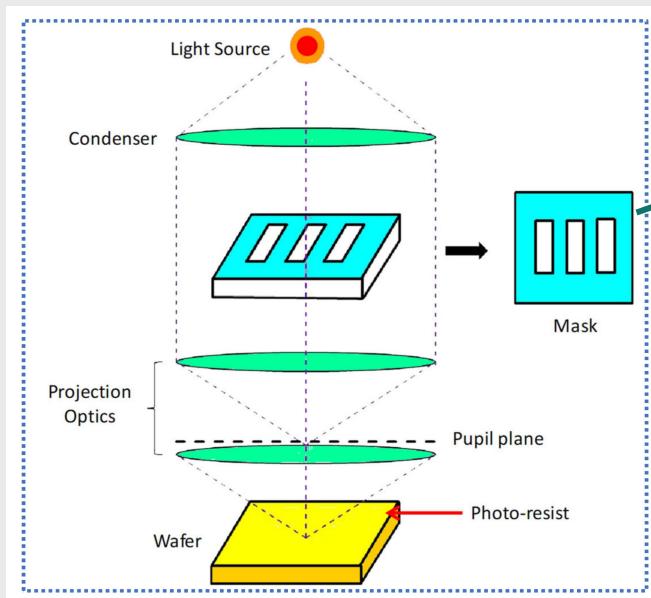
목표 웨이퍼 패턴에서 역산(Inverse Calculation)하여 최적의 마스크 형상 도출

직각 기반 단순 설계 X 수학적 최적화 알고리즘으로 계산된 복잡한 마스크 형상 사용

마스크를 의도적으로 왜곡하여 노광시 발생하는 물리적 오차를 선제적으로 상쇄

→ 대규모 물리 시뮬레이션 & 반복 최적화 필요

# LithoBench Dataset



## Optical Lithography (DUV/EUV) 공정 진행

- 빛의 회절(Diffraction) 및 공정 변수로 인한 패턴 왜곡 발생
- 설계된 Target 이미지와 실제 웨이퍼 패턴 간 불일치 (Process Variation)
- ⇒ **Reference ILT (Inverse Lithography Technology)** 알고리즘을 수행하여 마스크 최적화

이 과정을 다양한 회로 레이아웃에 반복 수행하여 LithoBench 데이터셋 구축

# Datasets

### LithoBench: Benchmarking AI Computational Lithography for Semiconductor Manufacturing

Su Zheng<sup>1</sup> Haoyu Yang<sup>2</sup> Binwu Zhu<sup>1</sup> Bei Yu<sup>1</sup> Martin D.F. Wong<sup>3</sup>

<sup>1</sup>The Chinese University of Hong Kong

<sup>2</sup>nVIDIA, Austin, USA

<sup>3</sup>Hong Kong Baptist University

#### Lithography Dataset ( $\approx 14$ GB)

- [데이터셋 카테고리] (MetalSet, StdMetal, StdContact, ViaSet 등)
- 1. 설계 및 입력 (Design & Input)
  - target/ # 우리가 원하는 최종 패턴 (Ground Truth)
- 2. 마스크 최적화 (Mask Optimization / ILT)
  - pixelILT/ # Pixel 기반 역공학 리소그래피(ILT) 결과
- 3. 공정 시뮬레이션 결과 (Simulation Output)
  - litho/ # Aerial Image (광학 이미지)
  - resist/ # Photoresist Profile (PR 현상 후)
  - printed/ # 최종 웨이퍼 위의 패턴 (Final Etch/Print)

## 1. 도입 배경 및 기술적 한계(Computational Bottleneck)

반도체 데이터는 기업의 핵심 지식재산(IP)으로 **외부 접근이 불가능 & 학습 데이터 부족**으로 인한 AI 연구의 진입 장벽 존재

기존 광학 근접 보정(OPC) 및 공정 보정 기술(Traditional ILT)은 정확하나 **연산 비용과 시간 소모가 막대** → AI(DL) 도입 필요성 대두

## 2. 의의

학계 최초의 대규모 공개 벤치마크 (120,000+ pairs), **폐쇄적인 리소그래피 분야의 연구 접근성 확보** 및 AI 성능 표준화

# Dataset Overview

## Data Structure

: target/printed/litho/resist/pixelILT

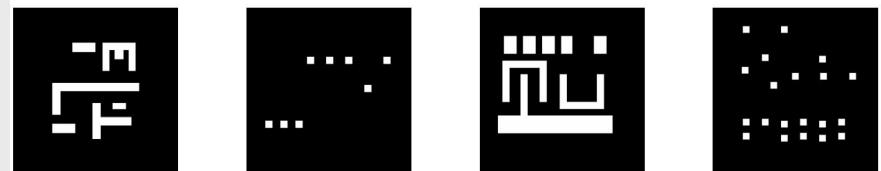
**MetalSet** : 16,472 images (per stage)

**StdMetal** : 271 images (per stage)

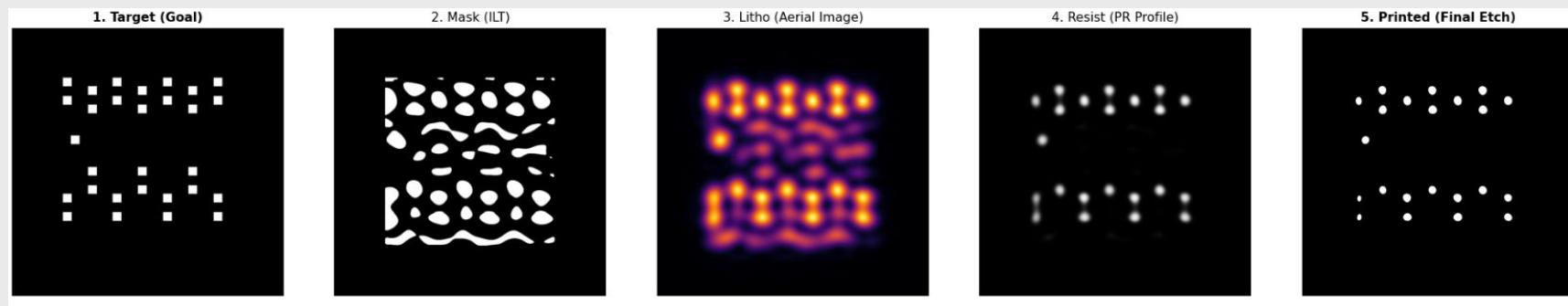
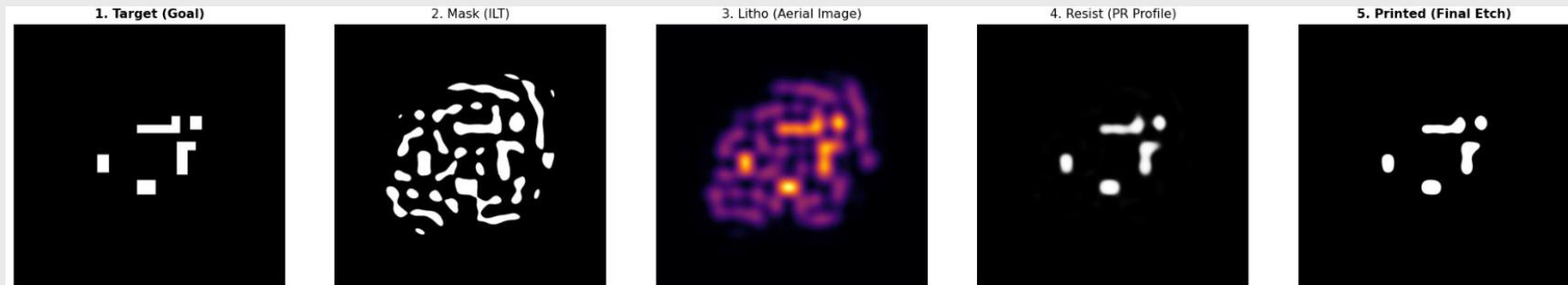
**StdContact** : 163 images (per stage)

**StdContactTest** : 165 images (per stage)

**ViaSet** : 116,414 images (per stage)



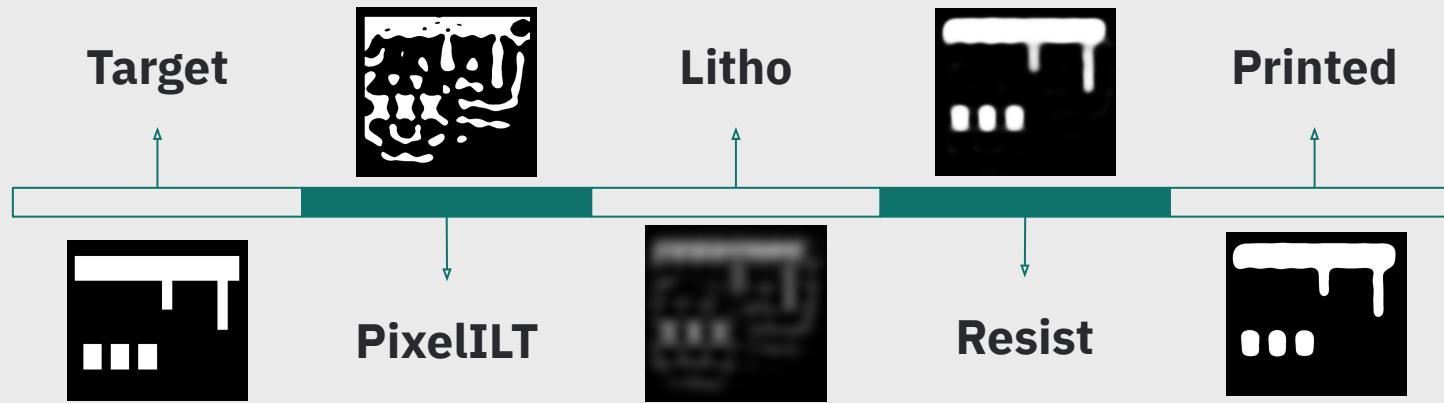
# Dataset Overview: Metal VS Via



02

## **Data Feature Analysis : StdMetal VS StdContact**

# Lithography 공정에서 에러는 어디서 결정되는가?



<다단계 비선형 공정>

→ 최종 오차 Printed-Target mismatch는 누적오차의 결과

RQ1. 최종 오차율에 대해 공정 단계별 기여도는 얼마인가?

RQ2. 공정 오차율은 마스크의 어떤 기하학적 특징과 연관되는가?

# Data & Pipeline

## Dataset

Lithography multi-stage mask images in StdMetal and ContactSet(StdContact+Test) Samples: 271 masks and 328 masks (filename-based exact matching)

## Error Metrics

XOR: Area-based mismatch

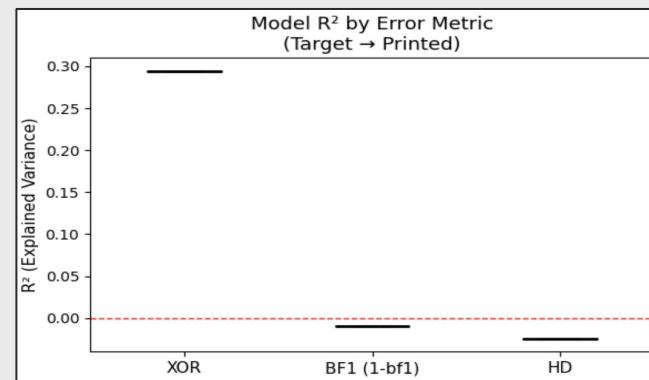
Boundary-F1 ( $1 - \text{bf1}$ ): Boundary accuracy

Hausdorff distance: Maximum boundary deviation

→ XOR was selected as the final error metric

## Preprocessing

Grayscale normalization → binary mask  
Target/PixelILT/Printed: Otsu threshold  
Litho/Resist: fixed threshold( $t=0.40$ )  
Morphological open/close for noise cleanup



## RQ1. 공정단계별 에러 기여도(StdMetal)

### Shapley value-based R<sup>2</sup> contribution analysis

#### Model setup

Response: xor\_target\_printed

Predictors:

- xor\_target\_pixelILT
- xor\_pixelILT\_litho\_t0.40
- xor\_litho\_resist\_t0.40

#### Uncertainty estimation

Bootstrap (300 runs)

Report:

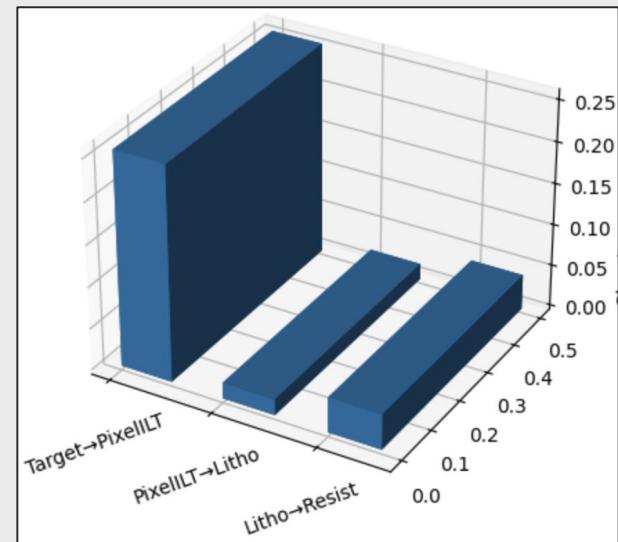
- median contribution
- 95% confidence interval
- contribution share (%)

# RQ1. 공정단계별 에러 기여도(StdMetal)

## Results:

상위 3개 공정

Process stage	Median Contribution	Share(%)	95% CI
Target → PixelILT	0.257	80.6%	[61.3, 93.6]
Litho → Resist	0.0419	13.4%	[3.5, 26.5]
PixelILT → Litho	0.0190	5.9%	[2.2, 12.0]



## RQ2. 기하학적 특징별 에러 기여도(StdMetal)

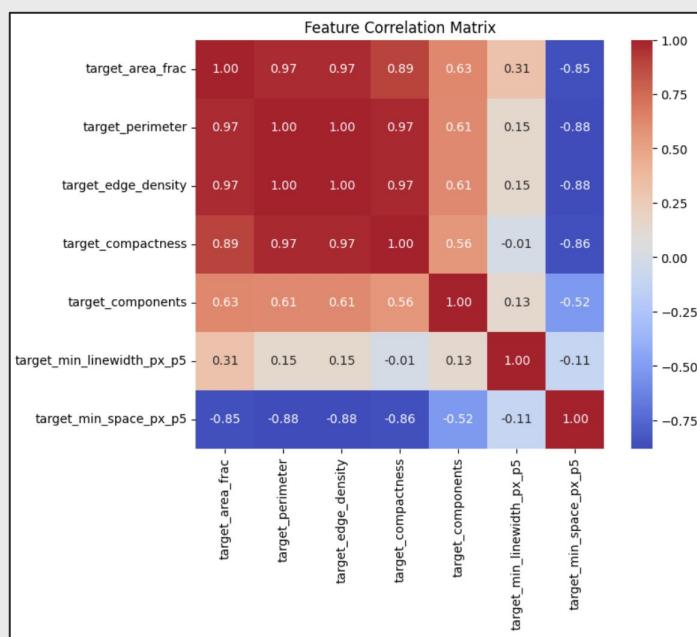
### Extracted Target Geometry Features

N=7

target_edge_density	Edge complexity
target_perimeter	Boundary length
target_compactness	Shape compactness
target_area_frac	Pattern coverage
target_minspace_px_p5	Minimum spacing (litho hotspot)
target_min_linewidth_px_p5	Minimum linewidth (critical dimension)
target_components	Disconnected components

# RQ2. 기하학적 특징별 에러 기여도(StdMetal)

## Extracted Target Geometry Features



Feature	VIF
target_edge_density	1,605,559.94
target_perimeter	1,591,762.13
target_compactness	915.19
target_area_frac	766.98
target_minspace_px_p5	4.51
target_min_linewidth_px_p5	2.10
target_components	2.05

## RQ2. 기하학적 특징별 에러 기여도(StdMetal)

### Extracted Target Geometry Features

N=3

#### 1. TOPOLOGICAL

→ target\_components

#### 2. LITHO-LIMIT

→ target\_min\_linewidth\_px\_p5, target\_min\_space\_px\_p5

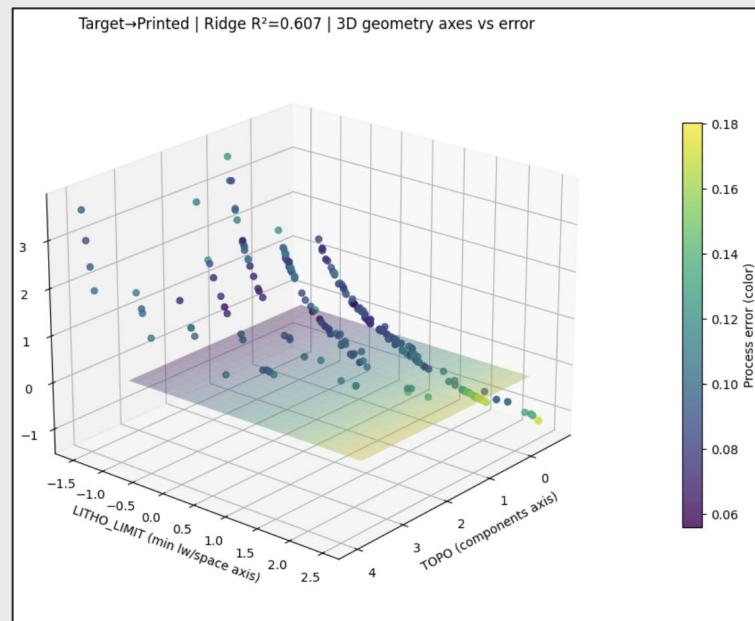
#### 3. STRUCTURAL DENSITY

→ target\_area\_frac, target\_perimeter, target\_edge\_density,  
target\_compactness

## RQ2. 기하학적 특징별 에러 기여도(StdMetal)

Method: Spline-Ridge R<sup>2</sup>

Results:



Group_features	Contribution_delta R <sup>2</sup>	Share(%)
Structural(z)	0.45	74.8
Litho_limit(y)	0.09	16.2
Topo(x)	0.05	8.8

→ 가장 위험한(오차율이 높은) 마스크

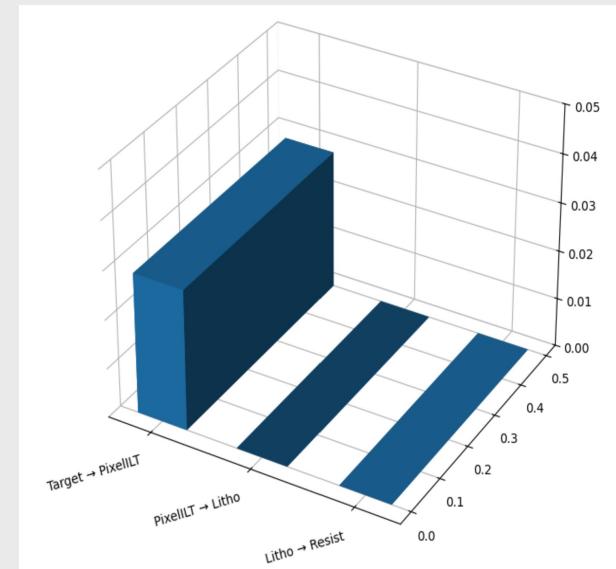
:TOPO ↑ + LITHO\_LIMIT ↓ + STRUCTURAL ↑

# RQ1. 공정단계별 에러 기여도(ContactSet)

## Results:

상위 3개 공정

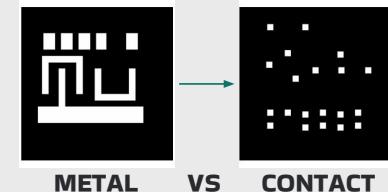
Process stage	Median Contribution	Share(%)	95% CI
Target → PixelILT	<b>0.0287</b>	<b>100.0%</b>	[0.015, 0.045]
Litho → Resist	0	0.0%	[0.0, 0.0]
PixelILT → Litho	0	0.0%	[0.0, 0.0]



## RQ2. 기하학적 특징별 에러 기여도(ContactSet)

### Extracted Target Geometry Features

N=7

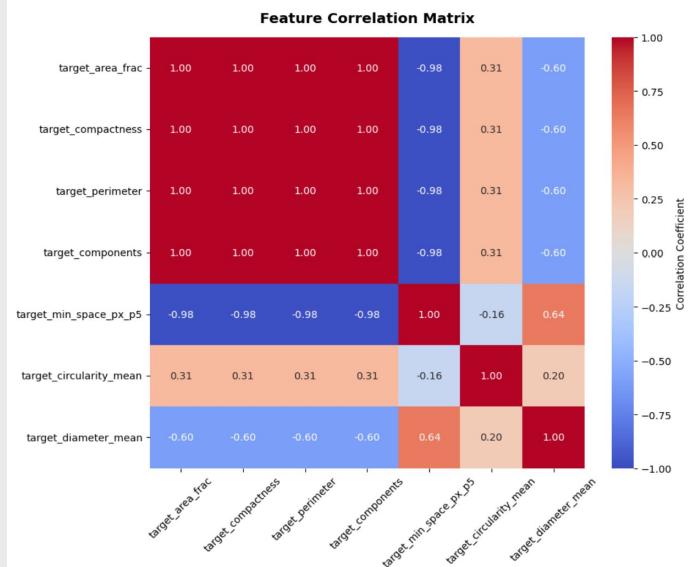


<code>target_area_frac</code>	Pattern Density (Coverage)
<code>target_compactness</code>	Shape compactness
<code>target_perimeter</code>	Total Boundary Length
<code>target_components</code>	Disconnected components
<code>target_min_space_px_p5</code>	Minimum spacing (litho hotspot)
<code>target_circularity_mean</code>	Via Roundness (Contact Quality)
<code>target_diameter_mean</code>	Critical Dimension (CD)

## RQ2. 기하학적 특징별 에러 기여도(ContactSet)

### Extracted Target Geometry Features

Extracted Target Geometry Features (Via-Specific)



Feature	VIF
target_area_frac	89,756,051,247.02
target_compactness	33,140.18
target_perimeter	8568.27
target_components	3709.67
target_min_space_px_p5	24.46
target_circularity_mean	3.34
target_diameter_mean	1.59

## RQ2. 기하학적 특징별 여러 기여도(ContactSet)

### Extracted Target Geometry Features

N=3

#### 1. Spacing (Layout Limit)

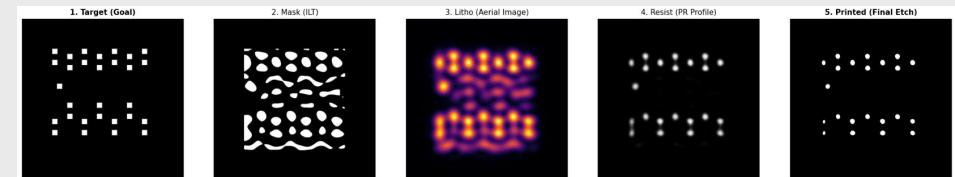
- target\_min\_space\_px\_p5
- target\_area\_frac, target\_components, target\_perimeter, target\_compactness, target\_edge\_density, target\_min\_linewidth\_px\_p5

#### 2. Dimension (Diameter)

- target\_diameter\_mean

#### 3. Form (Shape Fidelity)

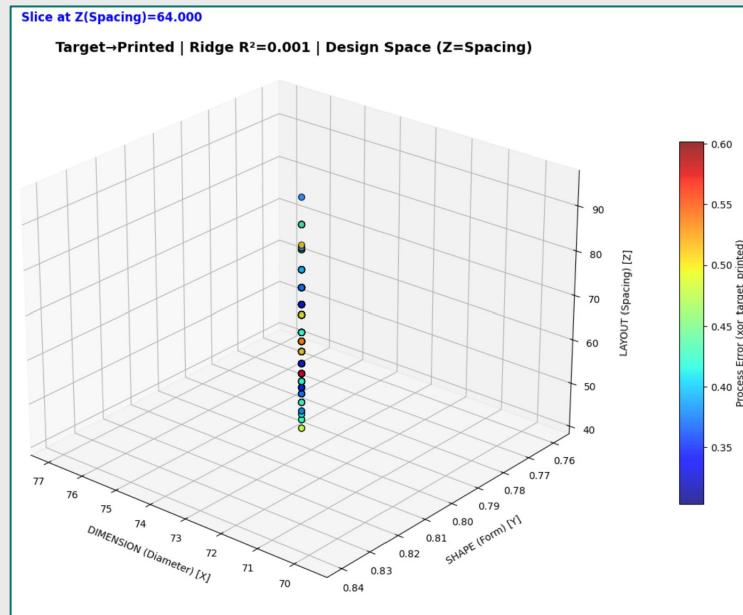
- target\_circularity\_mean



## RQ2. 기하학적 특징별 에러 기여도(ContactSet)

Method: Spline-Ridge R<sup>2</sup>

Results:



Group_features	Contribution_delta R <sup>2</sup>	Share(%)
Spacing(z)	0.1229	94.4
Form(y)	0.0047	3.6
Dimension(x)	0.0026	2.0

→ 가장 위험한(오차율이 높은) 마스크

: Form ↑ + Spacing ↓ + (Dimension ↑)

**03**

## **Price Perspective : Via VS Metal**

# Shot Count

Photolithography(노광) 공정에서 마스크 회로 패턴을  
웨이퍼 위에 노광하기 위해 필요한 빛의 노출(flash) 횟수의 총합

$\propto$  시간, 비용

낮을수록 고효율

→ Metal과 Via 데이터를 Shot Count 중심으로 분석해 보자!

# Shot Count Proxy

그러나, Shot count data의 부재...

PNG는 압축 파일 포맷

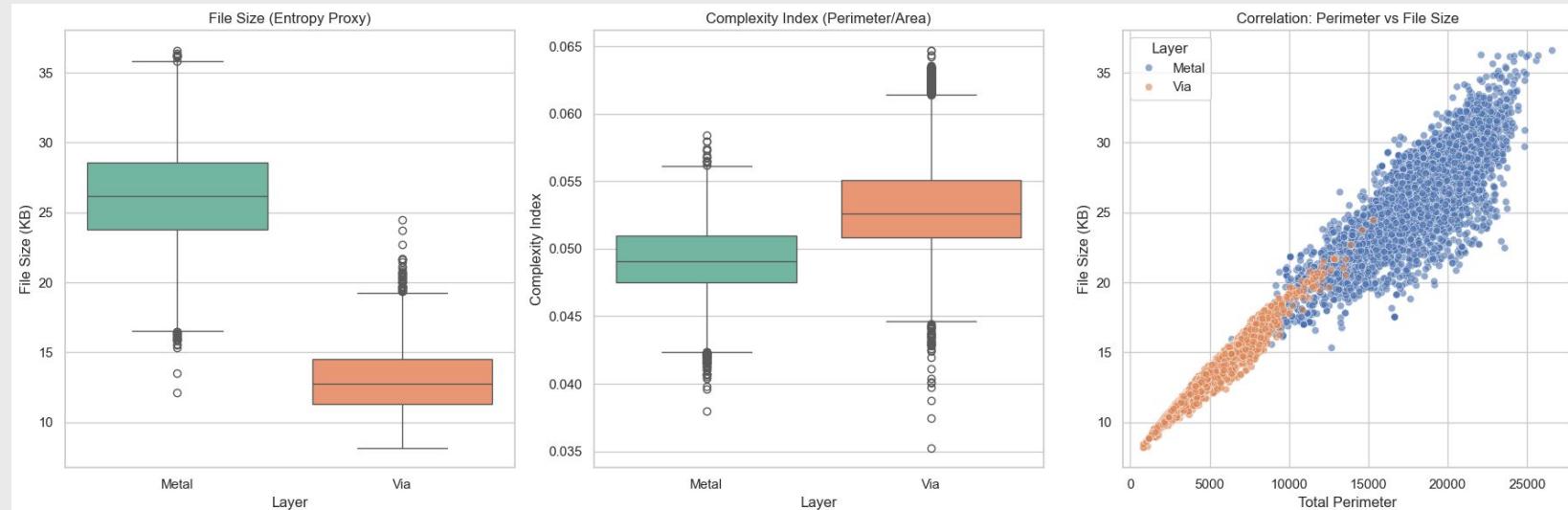
복잡한 파일일수록  
압축이 어려워  
파일 크기 증가

Mask writer는  
샷마다 직사각형을 프린트  
즉 곡선 프린트 어려움

마스크에 곡선이 많을수록  
∞ 샷 카운트 증가

→ 파일 크기(PNG), 둘레의 길이를 Proxy로 활용

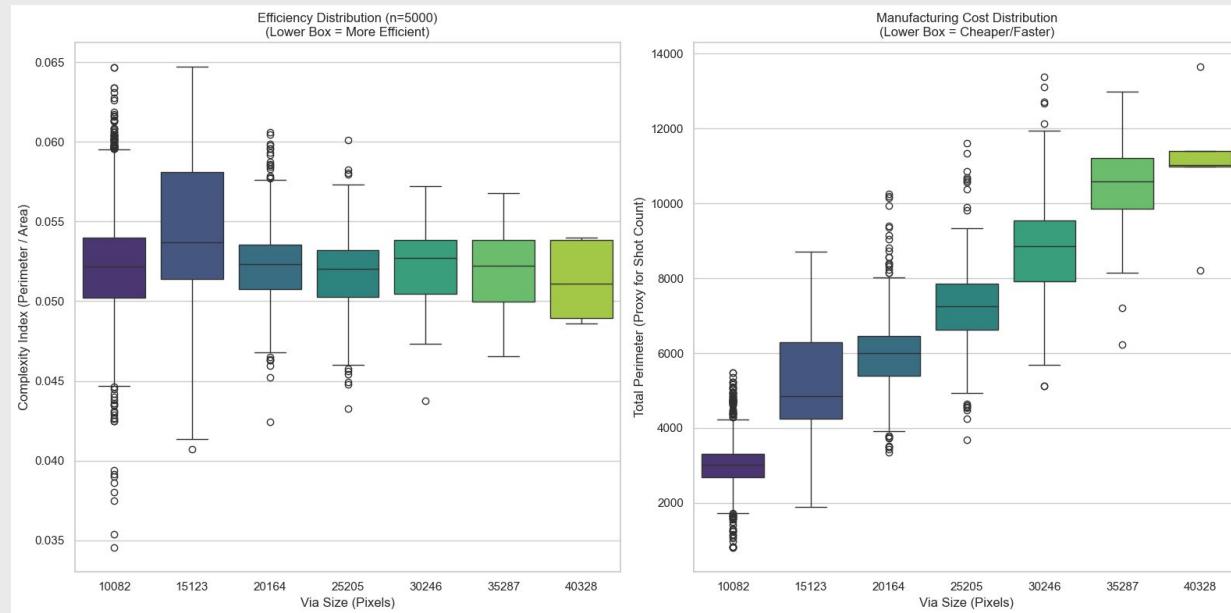
# Shot Count Proxy



Metal이 Via에 비해 파일 크기 ↑

Via가 Metal에 비해 (둘레 길이) / 면적 값 ↑ → 비효율적

# How can we make Via more efficient?



Via main Feature의 수가 증가함에 따라 복잡도는 유사, Shot Count 증가

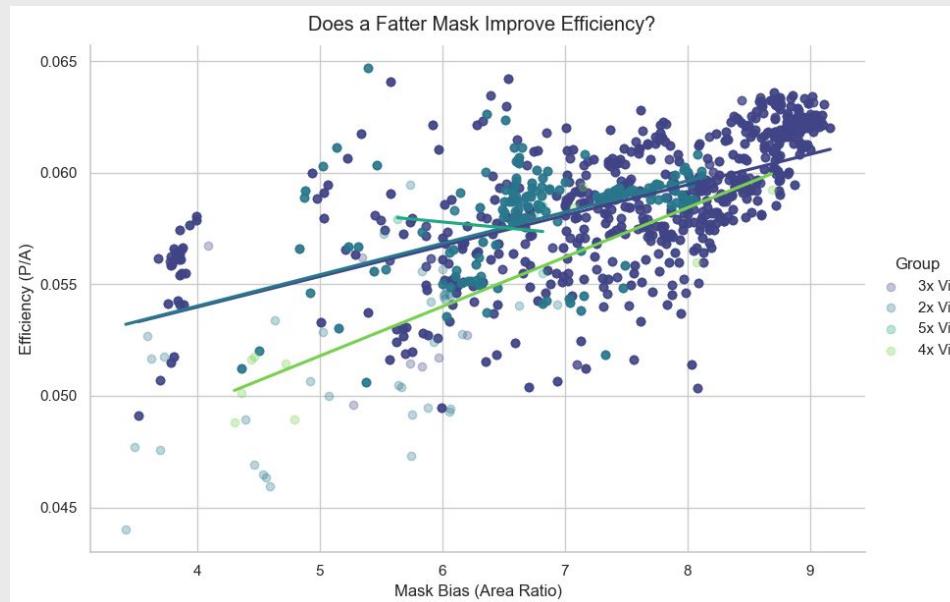
# How can we make Via more efficient?

What if we increase the mask feature size?

$$\text{마스크 bias} = \frac{\text{Mask Area}}{\text{Target Area}}$$

즉, “얼마나 많은 잉여 픽셀이 사용되었는가?”

# How can we make Via more efficient?



Mask bias가 증가하면 비효율적이어지고, 비용도 증가함

# IoU and SRAF Load Analysis

## IoU(Intersection over Union)

: 두 영역이 얼마나 겹쳐 있는가?  
→ Mask가 Target과 얼마나 유사한지

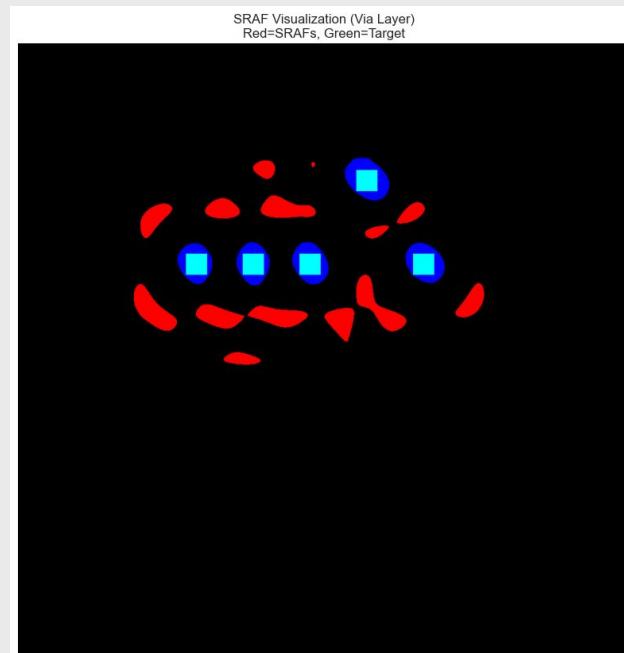
## SRAF(Sub-Resolution Assist Features)

: Mask에 있는 작은 Feature들  
웨이퍼에 프린트되지는 않으나  
포토리소그래피 공정의 품질을 높여줌

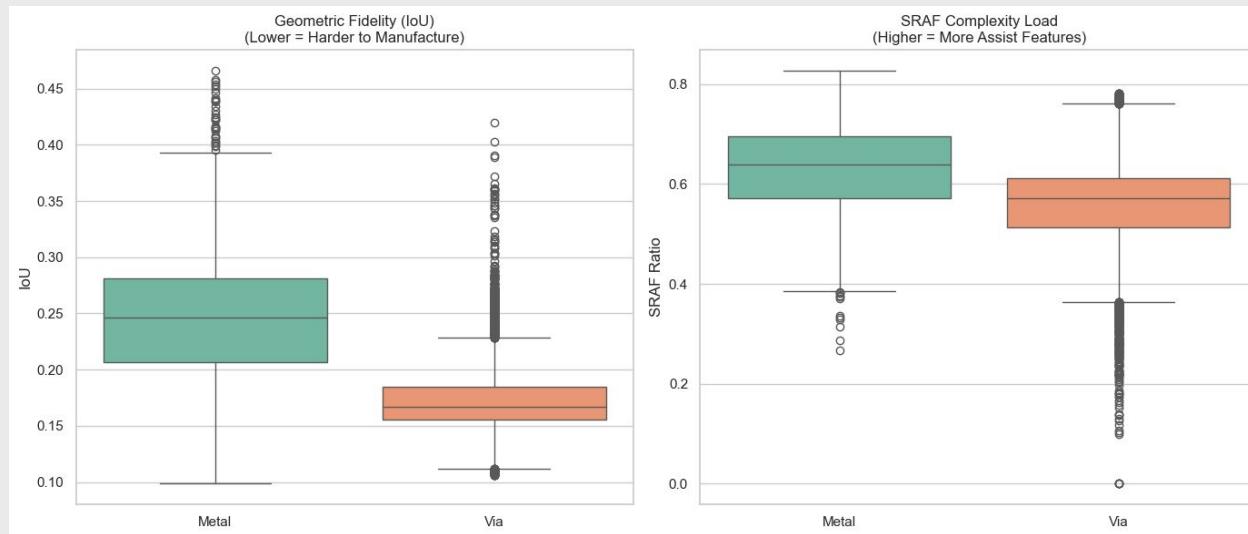
### SRAF 분류

: Connected Components Labeling  
→ Target과 겹치면 Main Features, 아니면 SRAF

$$\text{SRAF Load} = (\text{SRAF 픽셀 수}) / (\text{전체 픽셀 수})$$



# IoU and SRAF Load Analysis



Via가 정확하게 프린트하기 더 어려움 (low fidelity)

Metal은 비교적 정확하게 프린트되나, 이를 위해 더 많은 SRAF(보조 무늬)를 필요로 함

04

## FFT-Based Efficient Viewing Method

# Start from Sampling

MetalSet은 16,472장으로, StdMetal(271장)과 직접 비교 불가

즉, 표본 수 자체가 다르기에 분포 비교가 왜곡될 위험 존재



**Sampling!**

# Sampling Process

**1.Binarize + Downsample + Flatten**

**2.PCA**

Solving the Curse of Dimensionality, Noise filtering  
65,536dim -> 363dim (Cumulative Explained Variance: 92%)

**3.UMAP**

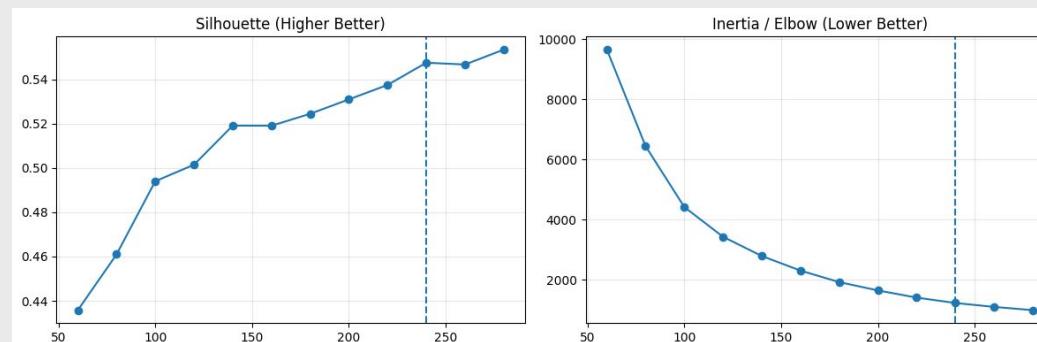
363dim -> 15dim

# Sampling Process

## 4. K-means Sweep

15차원 UMAP 공간에서 K를 후보군(60~280개)에 따라 반복 수행하며 Silhouette Score와 Inertia 계산 -> 최적의 K 도출

```
>>> Suggested bestK (simple) = 240
[Top candidates by silhouette]
      K  silhouette    inertia  min_cluster_size  zero_alloc_ratio
    240    0.547489  1220.761719            7        0.133333
    260    0.546676  1090.356812            7        0.153846
    280    0.553397   982.858948            7        0.185714
```

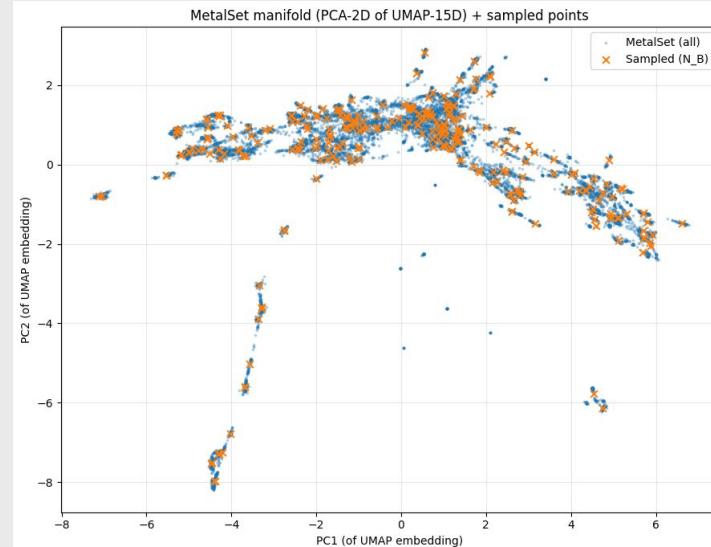


# Sampling Process

## 5. Stratified Prototype Sampling

최적화된 240개 군집에 대해  
각 군집의 데이터 개수에 비례하여  
샘플 수 할당

이후 각 군집 내에서 중심점과  
유클리드 거리가 가장 가까운 샘플  
271개 최종 선택



# Step1. Analysis of Layout complexity

- 복잡도 지표 정의

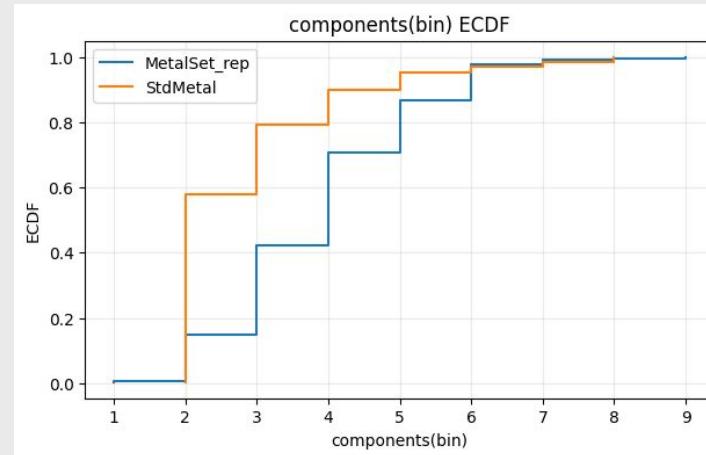
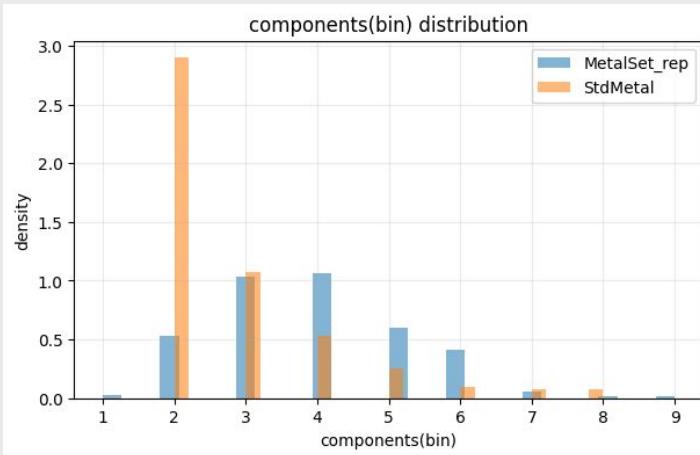
**Components:** 서로 떨어진 독립적인 패턴 덩어리의 수 : 구조 파편화

**Area\_frac:** 패턴 면적 비율 : 패턴 두께/채워진 정도

**Compactness:** 형태 복잡도( $\text{peri}^2/\text{area}$ ) : 경계 복잡도

# Step1. Analysis of Layout complexity

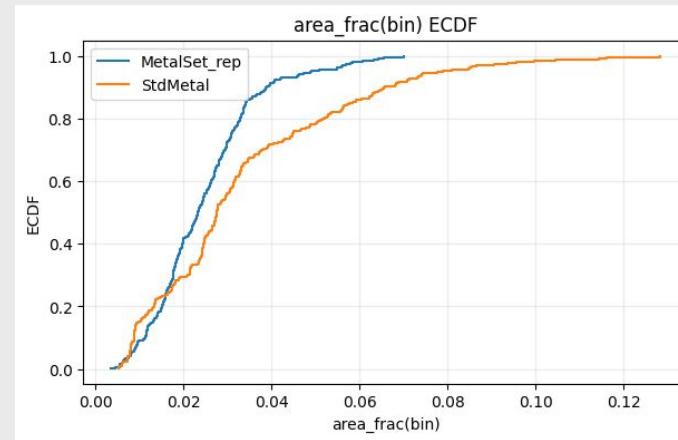
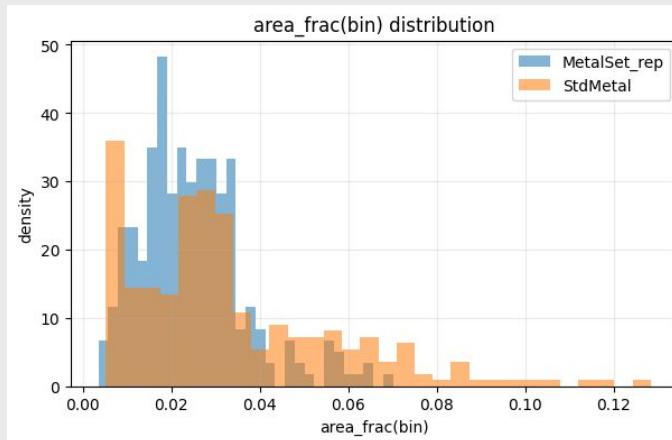
- Components



	Mean_metal	Mean_std	Diff_mean (metal-std)	p_mwu	p_ks	Cohen's d	Cliff's delta
<b>Components</b>	3.878	2.819	1.059	7.628015e-25	4.679689e-23	+0.809991	+0.492695

# Step1. Analysis of Layout complexity

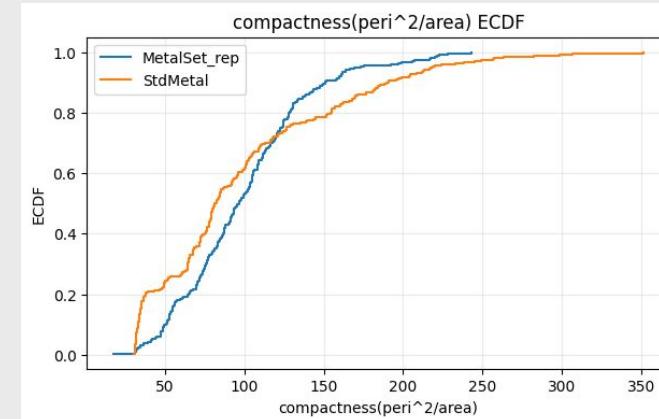
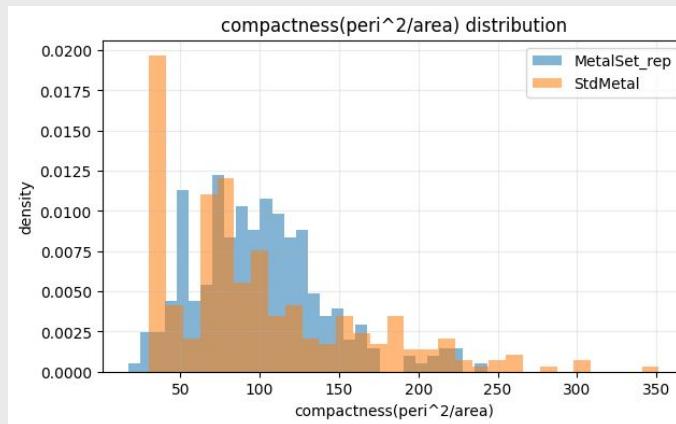
- Area\_frac



	Mean_metal	Mean_std	Diff_mean (metal-std)	p_mwu	p_ks	Cohen's d	Cliff's delta
Area_frac	0.024734	0.033547	-0.008813	1.629101e-04	1.772009e-05	-0.477004	-0.187198

# Step1. Analysis of Layout complexity

- Compactness



	Mean_metal	Mean_std	Diff_mean (metal-std)	p_mwu	p_ks	Cohen's d	Cliff's delta
Compactness	99.873319	99.497701	0.375618	2.371570e-02	7.940968e-04	0.007129	0.112294

# Step1. Analysis of Layout complexity

## Result of Step1

즉 MetalSet은 기하학적으로

파편화(Components ↑)되어 있고

더 얇으며(area\_frac ↓)

더 복잡(Compactness ↑)한

패턴들이 주를 이루고 있음

# Step1. Analysis of Layout complexity

## Result of Step1

즉 MetalSet은 기하학적으로

파편화(Components ↑)되어 있고

더 얇으며(area\_frac ↓)

더 복잡(Compactness ↑)한

패턴들이 주를 이루고 있음

=> 이 복잡도 차이가 printed 성능 차이로 이어질까?

# Step2. Target vs Printed Analysis

- 사용 지표 정의

$$\text{total\_iou}: \frac{TP}{TP+FP+FN}$$

**fp\_rate**: target에는 없는데 printed에 생긴 픽셀이 target 면적 대비 어느 정도인지 나타내는 지표

$$\frac{FP}{ref\_area}$$

**fn\_rate**: target에는 있는데 printed에 빠진 픽셀이 target 면적 대비 어느 정도인지 나타내는 지표

$$\frac{FN}{ref\_area}$$

**area\_error**: printed 면적이 target 대비 얼마나 차이나는지 나타내는 지표

$$\frac{|pred\_area-ref\_area|}{ref\_area}$$

\*ref\_area: target에서 1(존재)인 픽셀 수

\*pred\_area: printed에서 1(존재)인 픽셀 수

## Step2. Target vs Printed Analysis

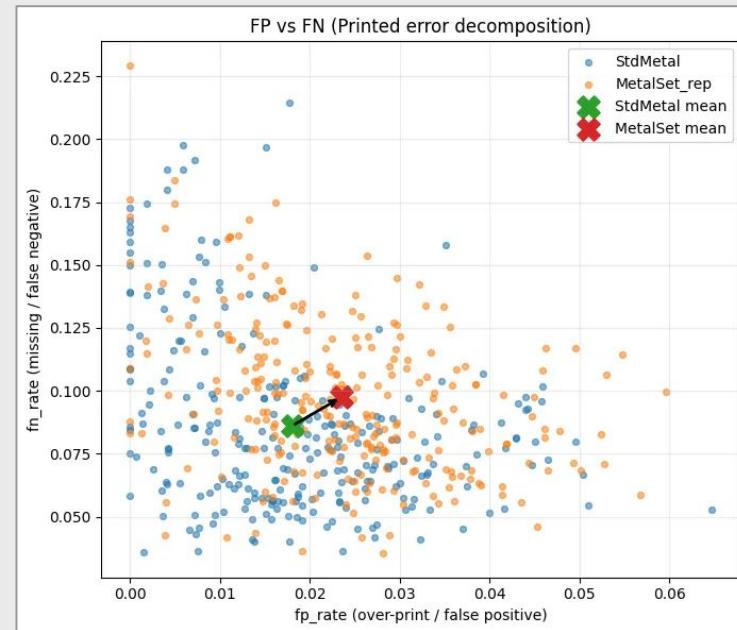
	Mean_metal	Mean_std	Diff_mean (metal-std)	p_mwu	p_ks	Cohen's d	Cliff's delta
total_iou	0.881923	0.897539	-0.015616	<b>9.125869e-12</b>	<b>4.410409e-11</b>	-0.518229	-0.338571
fn_rate	0.097369	0.086222	0.011147	<b>1.089500e-07</b>	<b>3.123359e-06</b>	0.340772	0.263681
fp_rate	0.023470	0.018084	0.005386	<b>1.877967e-07</b>	<b>4.876566e-06</b>	0.436152	0.258697
area_error	0.073900	0.068227	0.005673	4.501447e-03	7.940968e-04	0.146304	0.141038

# Step2. Target vs Printed Analysis

## Result of Step2

MetalSet은 StdMetal 대비  
Printed 성능이 유의하게 낫다.

그 하락은  
과인쇄(FP)와 미인쇄(FN)의  
동시 증가로부터 도출된 것

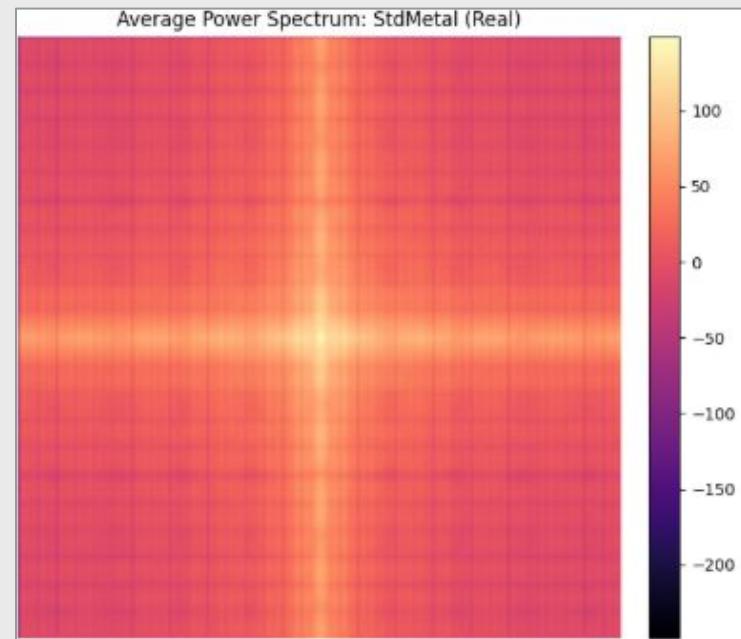


## Step3. Analysis of Spectral Entropy using FFT

### 주파수 도메인 변환

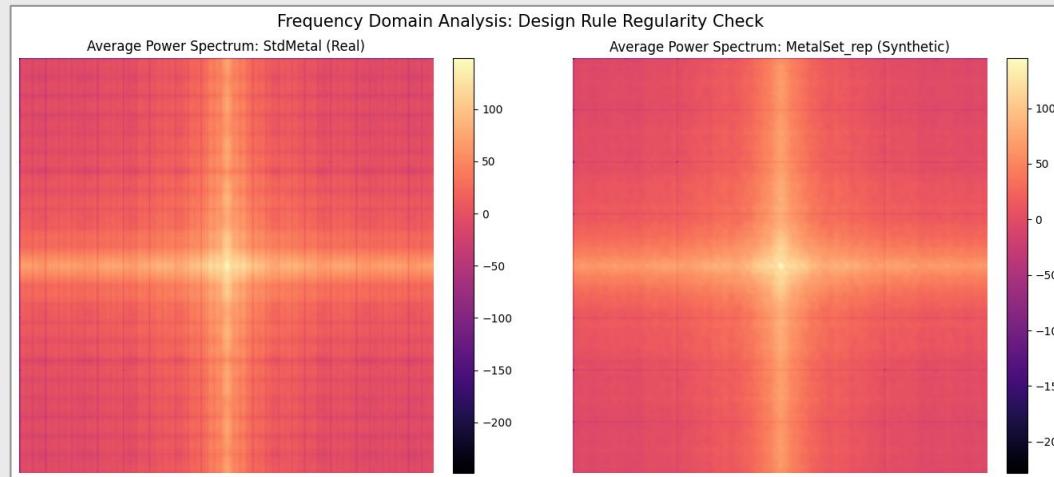
2D FFT(Fast Fourier Transform)를 활용하여 기하학적 레이아웃 이미지를 주파수 성분으로 변환

패턴의 물리적 주기성과 규칙성을 정량적으로 분석



# Step3. Analysis of Spectral Entropy using FFT

## StdMetal vs MetalSet: 주파수 영역의 격차



**분석 내용:** 2D FFT를 통해 실제 공정(Std)과 합성(Metal) 데이터의 평균 파워 스펙트럼을 산출하여 물리적 규칙성을 비교

**핵심 결과:** StdMetal(왼쪽)은 에너지가 특정 축에 집중된 규칙적 형태를 띠는 반면, MetalSet(오른쪽)은 에너지가 사방으로 분산된 Noise가 관찰됨

# Step3. Analysis of Spectral Entropy using FFT

## Stage-wise IoU, 에러의 발원지 추적

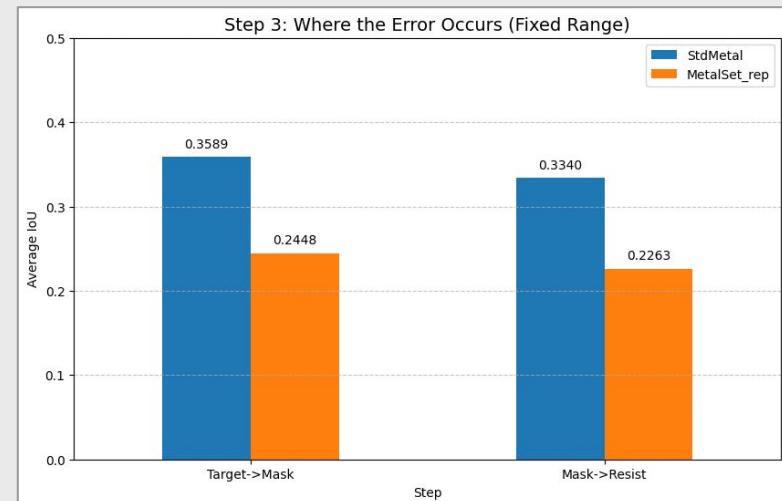
### 분석 내용

전체 공정을 설계( $T \rightarrow M$ )와 시뮬레이션( $M \rightarrow Z$ ) 단계로 나누어 어느 지점에서 성능 하락이 시작되는지 분석

### 핵심 결과

MetalSet은 첫 단계인 Target $\rightarrow$ Mask(ILT) 단계에서 이미 Std 대비 IoU가 급감하며, 이후 단계에서 에러가 누적

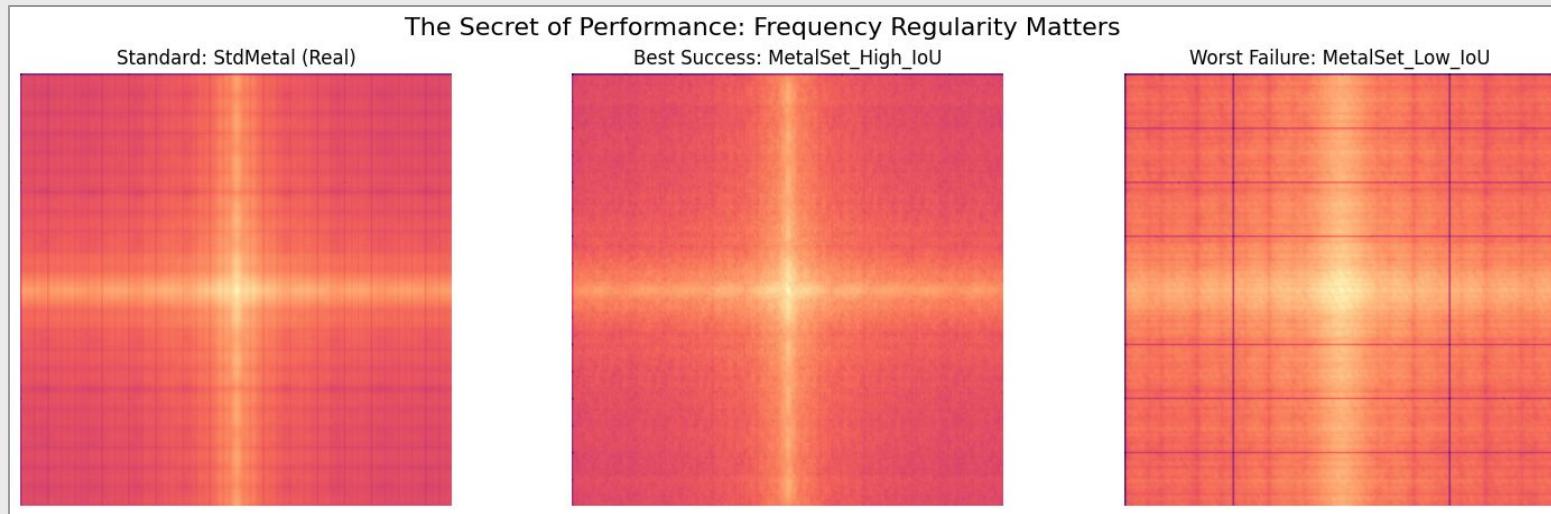
**초기 단계(Target $\rightarrow$ Mask)의 성능 붕괴:**  
MetalSet은 첫 단계부터 Std 대비 IoU가 급감



```
--- Average IoU Breakdown ---
StdMetal | T->M: 0.3589 | M->Z: 0.3340
MetalSet | T->M: 0.2448 | M->Z: 0.2263
```

# Step3. Analysis of Spectral Entropy using FFT

성능 상위(Best) 및 하위(Worst) 샘플의 주파수 특성 대조 분석



**Best set:** 합성 데이터임에도 불구하고 주파수 에너지가 특정 축에 집중되어 실제 공정(Std)의 디자인 룰을 잘 따르고 있음

**Worst set:** 주파수 에너지가 안개처럼 사방으로 분산된 현상이 심각하며, 이 무질서함이 모델의 예측을 방해함

# Step3. Analysis of Spectral Entropy using FFT

## Spectral Entropy, 무질서도의 정량화

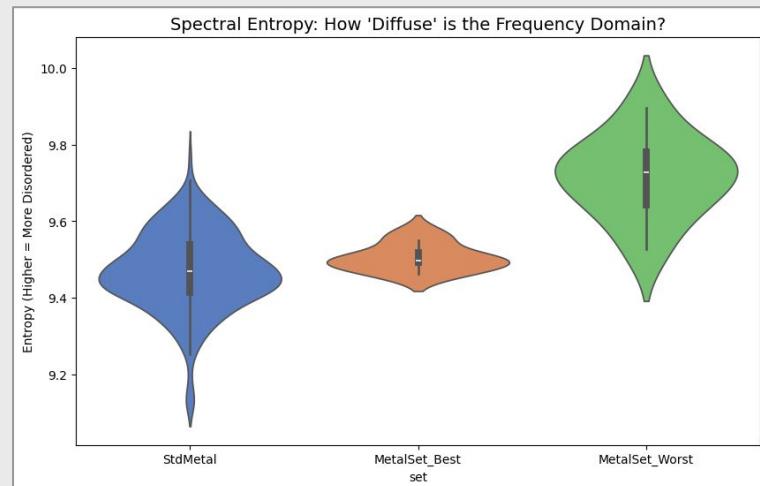
### 분석 내용

2D FFT 기반의 Spectral Entropy 분석을 통해 데이터셋별 주파수 무질서도를 정량화하고, 성능(IoU)과의 물리적 상관관계를 도출

### 핵심 결과

성능 하위 그룹(Worst)은 실제 공정(Std) 대비 압도적으로 높은 엔트로피를 기록

입력 데이터의 엔트로피는  
모델의 성능을 예측하는 선형 지표



# Step3. Analysis of Spectral Entropy using FFT

## 주파수 엔트로피 기반 품질 필터링을 통한 공정 신뢰도 회복

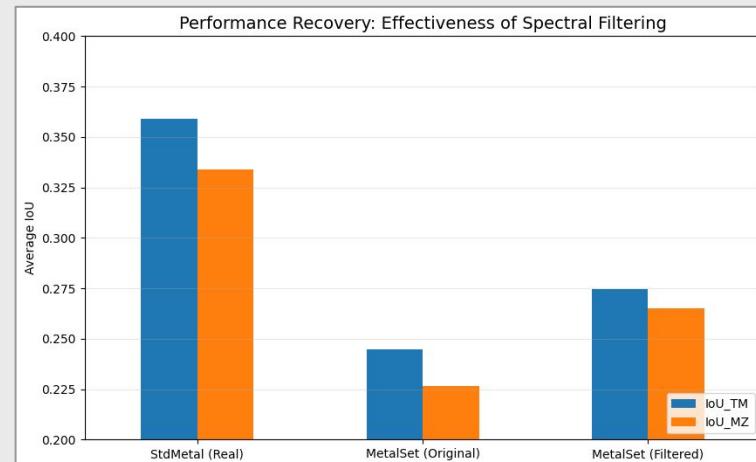
### 분석 방법

모델의 예측 결과(IoU)를 보지 않은 상태에서  
오직 주파수 엔트로피 수치가 높은 샘플들을  
데이터셋에서 제외한 후,  
남은 샘플들의 평균 성능을 재산출

### 핵심 결과

중앙의 MetalSet (Original)에 비해  
우측의 Filtered 의 데이터셋의 성능이 좋은 것을 확인

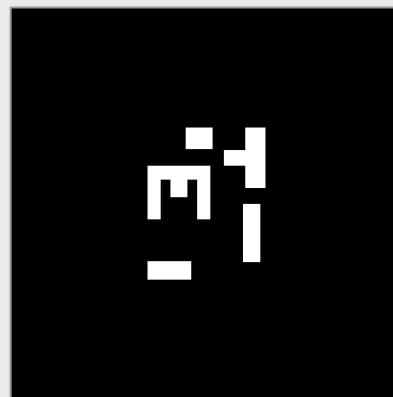
GPU 모델 없이 오직 주파수 연산만으로 확인 가능하기에  
실무적으로 매우 가치 있는 품질 관리 수단



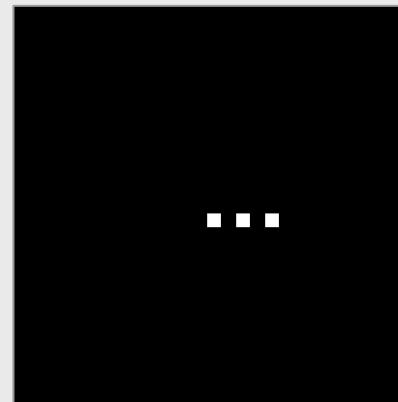
--- Improvement Summary ---  
Target->Mask IoU Improvement: 12.19%

## Step4. Comparative Analysis of ViaSet

Q. 패턴이 단순해지면(Vertical Via) 모델의 성능은 당연히 올라갈까?



MetalSet의 Target data

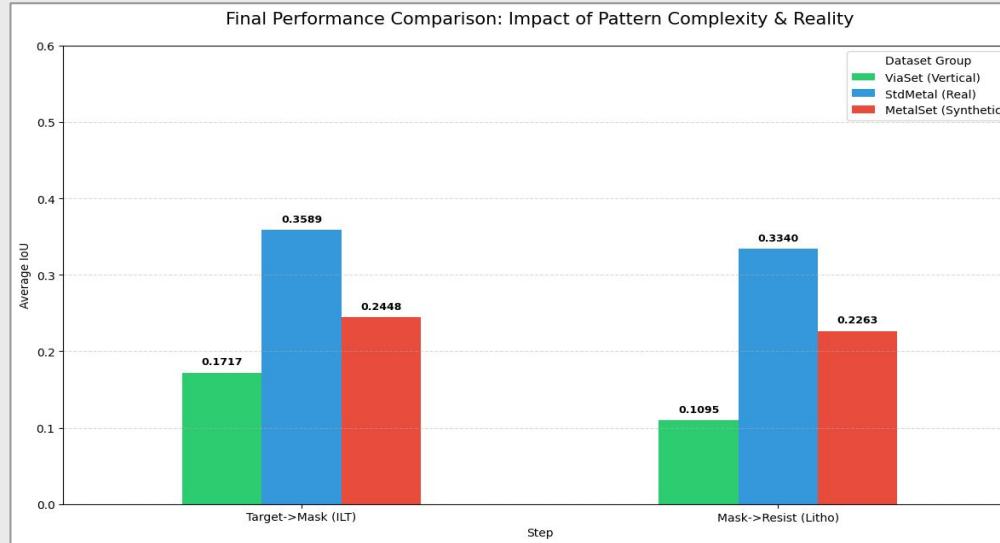


ViaSet의 Target data

수평 구조의 Metal 대비 기하학적으로 훨씬 단순한 수직 구조의 ViaSet(328개 샘플)을 추가  
-> 성능(IoU)측정과 FFT분석 등을 수행

# Step4. Comparative Analysis of ViaSet

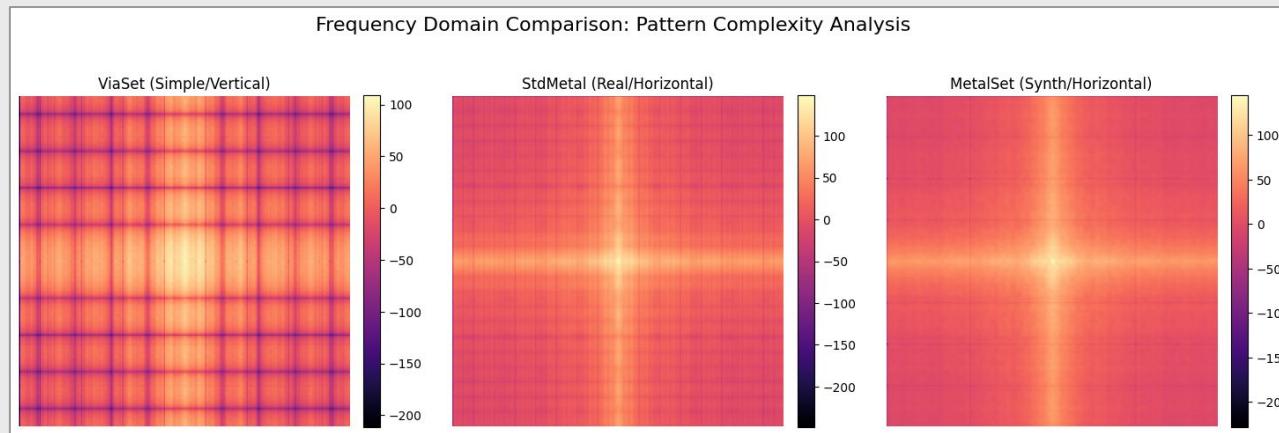
## IoU 성능 비교



**성능 불일치:** ViaSet은 시각적 단순함에도 불구하고 전 데이터셋 중 최저 IoU(약 0.17)를 기록

# Step4. Comparative Analysis of ViaSet

동일 분석 기법(FFT)을 통한 가설 검증



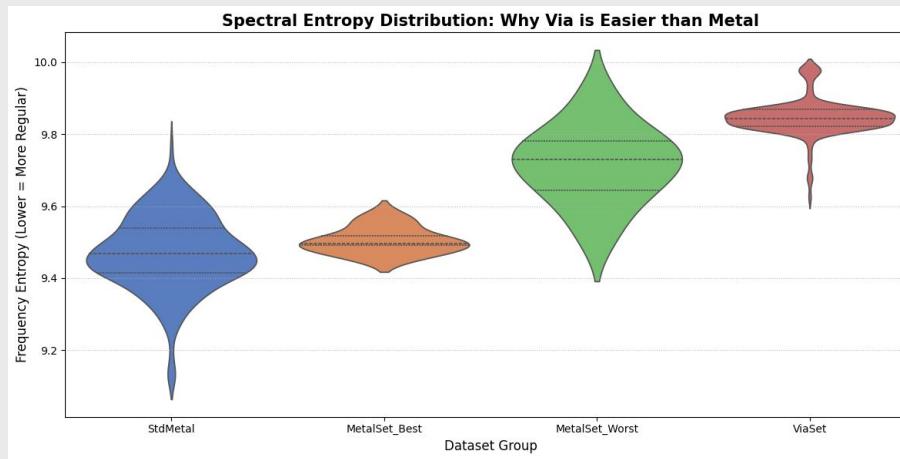
## 주파수 무질서도 확인

주파수 영역 분석 결과, ViaSet은 에너지가 축에 집중되지 못하고 전 영역으로 산란되어 MetalSet보다 높은 엔트로피를 가짐을 확인

아주 작은 점 하나를 FFT 하면, 에너지가 한곳에 모이지 못하고 전 주파수 대역으로 퍼지는 경향 존재

## Step4. Comparative Analysis of ViaSet

데이터셋별 Entropy 비교: 형태의 단순함을 압도하는 주파수 에너지 산란



### 엔트로피의 역설

시각적으로 가장 단순한 ViaSet이 실제로는 모든 그룹 중 가장 높은 주파수 엔트로피 분포를 보이며, 이것이 모델 성능 봉괴의 직접적 원인임을 입증

## Step5. 기하학적 복잡성과 주파수 도메인 성능 저하의 인과관계

### Result of Step1

즉 MetalSet은 기하학적으로

파편화(Components ↑)되어 있고

더 얇으며(area\_frac ↓)

더 복잡(Compactness ↑)한

패턴들이 주를 이루고 있음



### 기하학적 지표는 설계자가 제어할 수 있는 조절 변수

엔트로피가 아무리 중요해도 설계자가  
'엔트로피를 0.1 낮춰야지'하고  
설계할 수는 없음

엔트로피는 ‘최종적 물리 점수’

# Conclusion: A Holistic Approach for Lithography Optimization

## Data Feature Analysis

공정 내 오차 기여 : Target -> PixelILT 과정에서 Error 발생 비중 압도적

오차-기하학적 패턴 복잡성 : Metal -> 형상복잡도 & CD 복합적 작용, Via -> Spacing 압도적  
Layer 특성에 맞춘 차별화된 OPC/ILT 전략 필요

## Price Perspective

Via가 Metal에 비해 비용 측면에서 비효율적

## Methodology - Efficiency via FFT

Target 비교에서 Metalset(합성)이 StdMetal(실제)보다 기하학적 복잡 & printed 성능 ↓  
이러한 복잡성은 FFT를 통한 효율적 분석 가능

# Reference

1. Zheng, Su, et al. "Lithobench: Benchmarking ai computational lithography for semiconductor manufacturing." *Advances in Neural Information Processing Systems* 36 (2023)
2. Photolithography. (n.d.). ScienceDirect. Retrieved February 5, 2026, from <https://www.sciencedirect.com/topics/physics-and-astronomy/photolithography>
3. Dillinger, T. (2022, June 2). Inverse lithography technology – A status update from TSMC. SemiWiki.<https://sem/wiki.com/semiconductor-manufacturers/tsmc/313540-inverse-lithography-technology-a-status-update/>
4. Sharma, E., Rathi, R., Misharwal, J., Sinhmar, B., Kumari, S., Dalal, J., & Kumar, A. (2022). Evolution in lithography techniques: microlithography to nanolithography. *Nanomaterials*, figure 2
5. YouTube video (ID: ShYWUlj2FZs). <https://youtu.be/ShYWUlj2FZs>  
GIF via MakeAGif: <https://makeagif.com/amp/6ECBKo>

# Thank you!