DLS-DMO: Towards High Accuracy DL-Based OPC With Deep Lithography Simulator

Guojin Chen
supervised by
Prof. Yu Bei

Department of Computer Science & Engineering
The Chinese University of Hong Kong

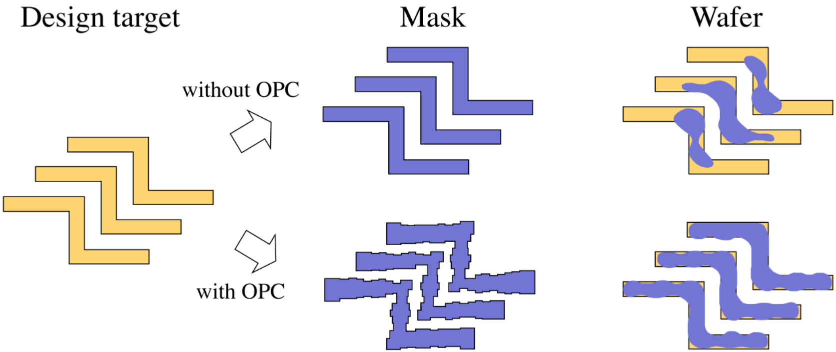
June 17, 2020

Outline

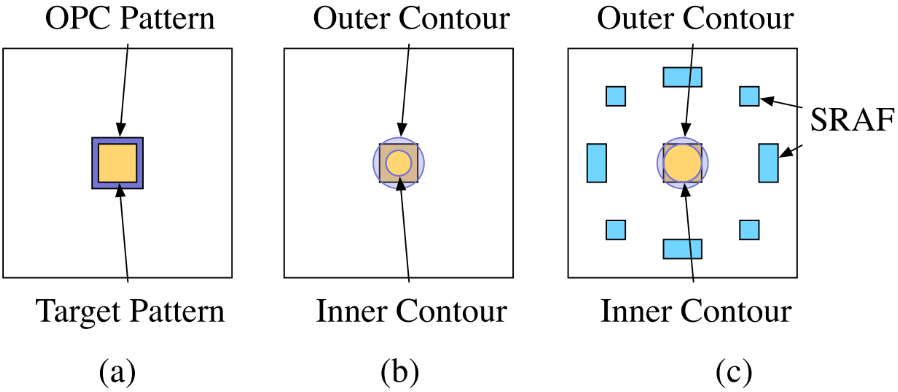
Background and problem formulation

Project backgroud

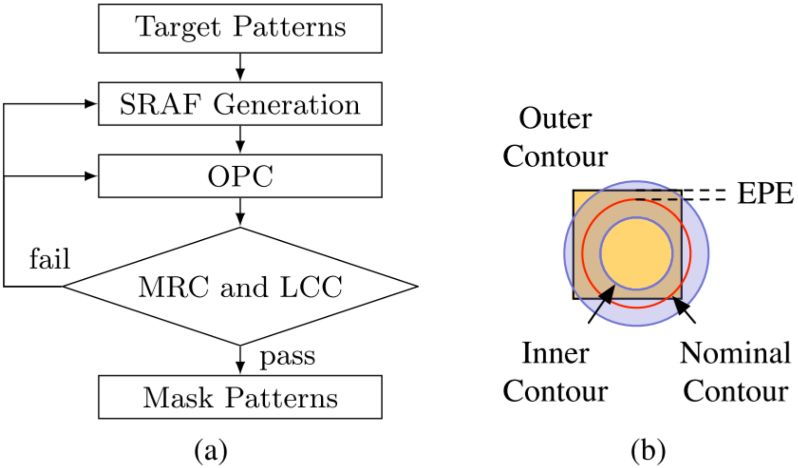
Optical proximity correction (OPC) is a photolithography enhancement technique commonly used to compensate for image errors due to diffraction or process effects.



PRELIMINARIES of OPC: DESIGN, SRAF, MASK, WAFER



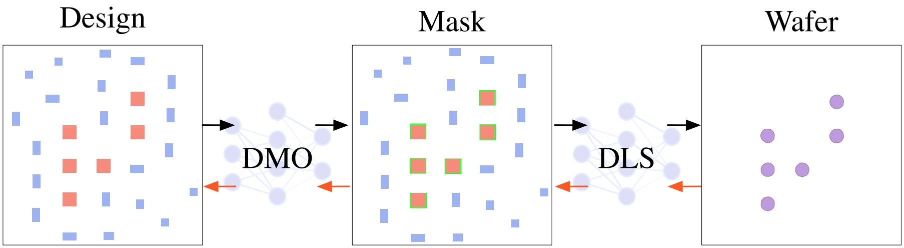
PRELIMINARIES of OPC: Flow, EPE, PVBand



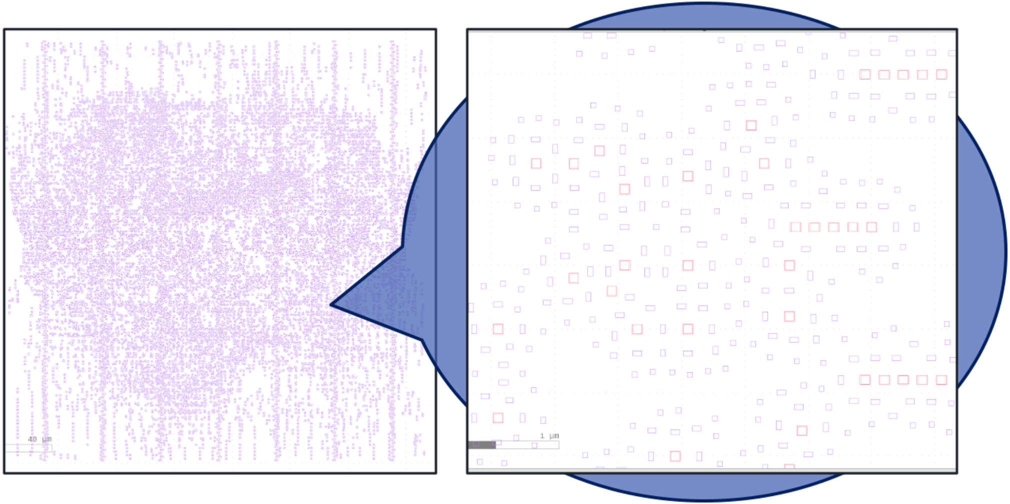
Goal: Using NN to simulate this process And beat one commercial products: Calibre

Two main step

OPC and Litho



Goal: Test our model on the industry data

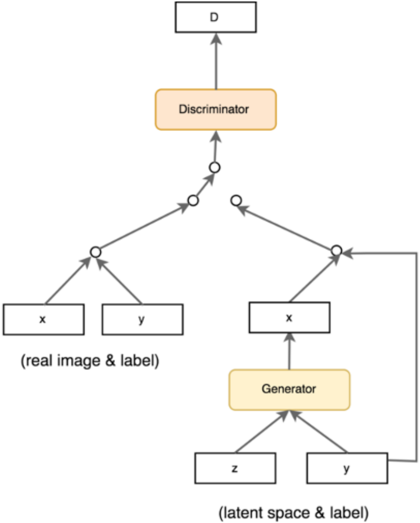


Outline

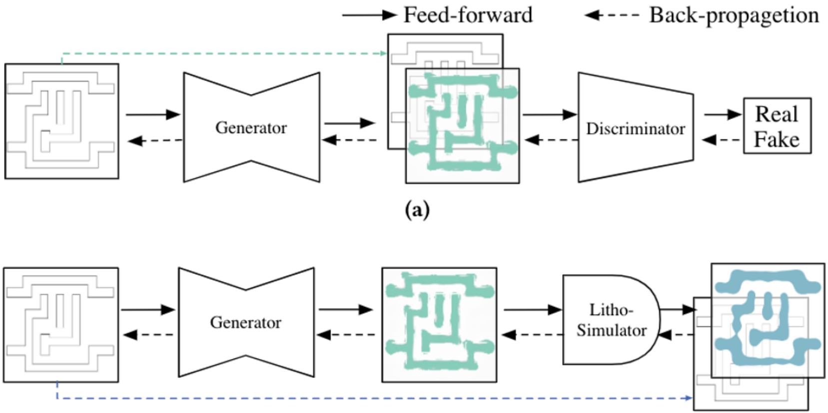
cGAN

Objective function

$$
\begin{aligned}
&\mathcal{L}_{cGAN}(G, D) \\
=&\mathbb{E}_{x,y}[\log D(x, y)] \\
+&\mathbb{E}_{x,z}[\log(1 - D(x, G(x, z)))].
\end{aligned} \tag{1}
$$



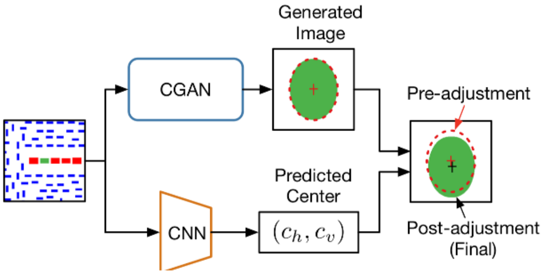
OPC stage previous work: GAN-OPC



GAN-OPC: shortages

- We cannot control the litho simulator.
- ILT-based model, come from MOSAIC, small layout.
- Only initial solution, bottleneck on the ILT-model.

Litho stage previous work: LithoGAN



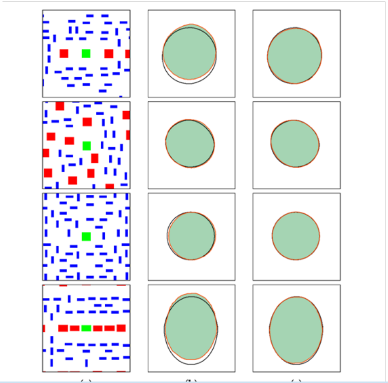
$$\mathcal{L}_{cGAN}(G,D) = \mathbb{E}_{x,y}[\log D(x,y)] + \mathbb{E}_{x,z}[\log(1 - D(x, G(x,z)))]$$

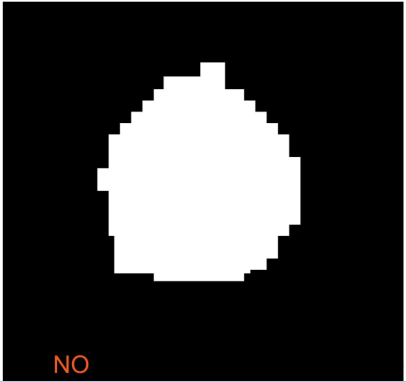
$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[\| y - G(x,z) \|_1]$$

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda \mathcal{L}_{L1}(G)$$

LithoGAN: shortages

1. Wafer may not have center. Did not make full use of cGAN.
2. The center shift is over design, we just need a powerful generator.
3. Mask must be at the center, one time can only generator one wafer(in the center, few in the dataset.)





Outline

Goal

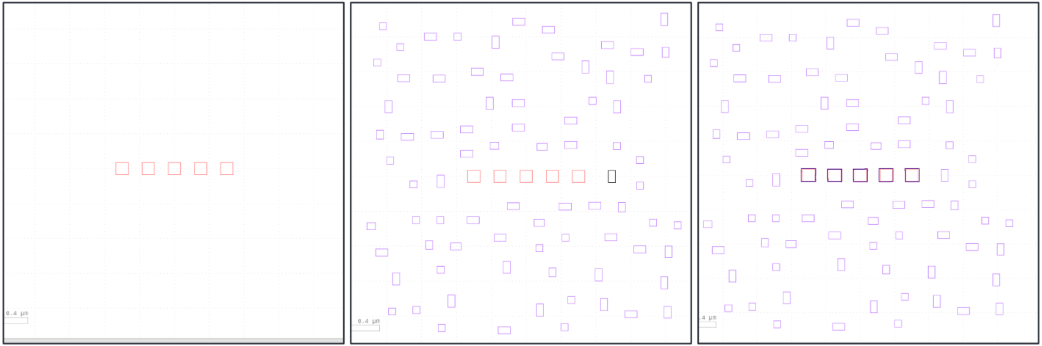
Problem

- Initial solution need further correction.
- One time one via lithography process.
- Low accuracy and small layout.

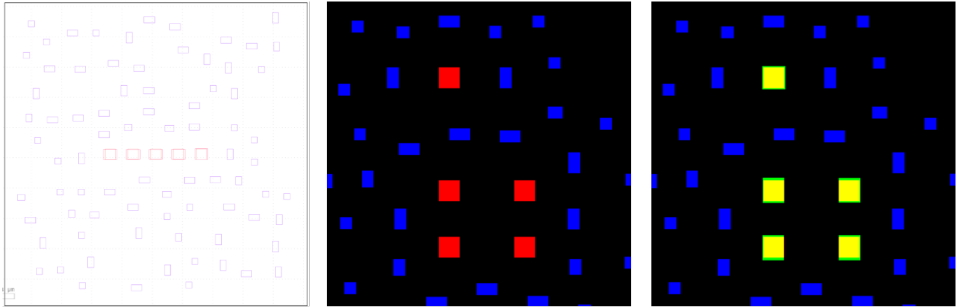
Solution: DLS-DMO

- End-to-end mask optimization without using traditional model.
- High resolution cGAN model.
- Window splitting algorithm for large layout.

Generate Training set



Self-generated datasets



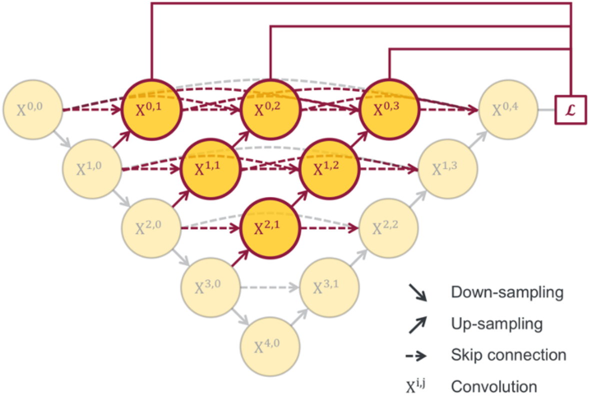
Layouts   Design   Mask

DCGAN-HD: solution for higher resolution

- Generator: DCUNet++
- Discriminator: Multi-discriminator
- Perceptual Losses

DCUNet++: Generator of DCGAN-HD

UNet++

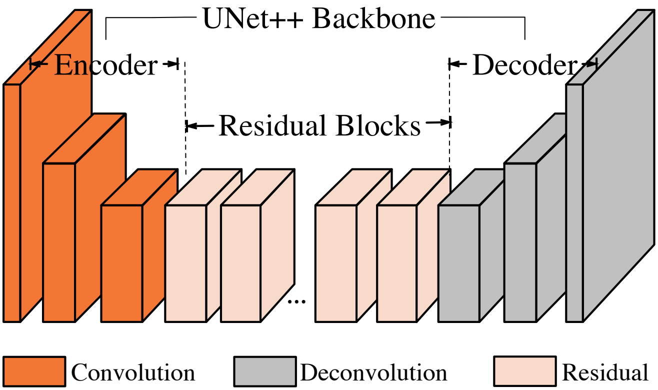


Arch.

- **UNet++ for low-level information.**
- Residual blocks

DCUNet++: Generator of DCGAN-HD

DCUNet++



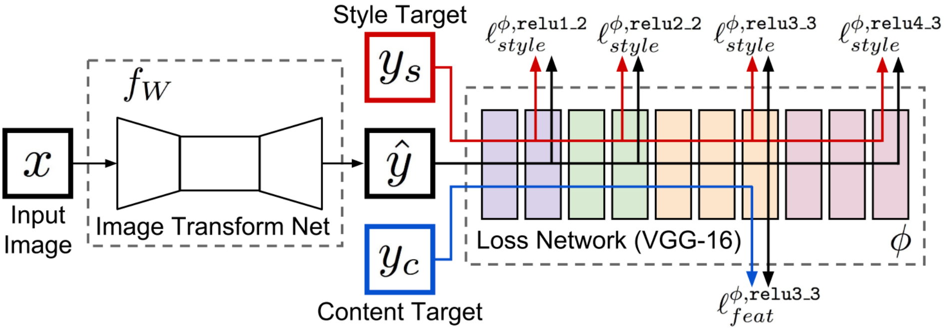
Arch.

- UNet++ for low-level information.
- **Residual blocks**

Multi Scale discriminator

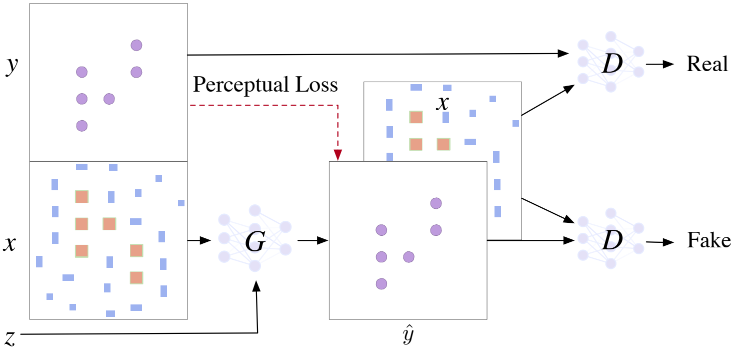
We design a multi-scale discriminator, different from pix2pixHD using 3 discriminators, our design uses 2 discriminators that have an identical network structure but operate at different image scales, which namedD1,D2. Specially, the discriminators D1,D2are trained to differentiate real and synthesized images at the 2 different scales, 1024×1024 and 512×512 respectively. As in pix2pixHD claimed, the multi-scale design helps the training of high-resolution model easier.

Perceptual Loss



DLS training

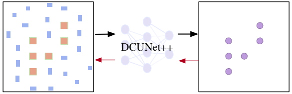
$$\mathcal{L}_{DLS} = \sum_{k=1,2} \mathcal{L}_{cGAN}(G, D_k) + \lambda_0 \mathcal{L}_{L_P}^{G,\Phi}(y, \hat{y}). \tag{2}$$



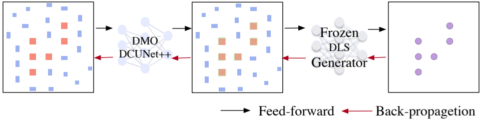
DMO training

$$\mathcal{L}_{DMO} = \sum_{k=1,2} \mathcal{L}_{cGAN}(G_{DMO}, (D_{DMO})_k) + \lambda_1 \mathcal{L}_{LP}^{G_{DMO},\Phi}(x, \hat{x}). \quad (3)$$

$$\mathcal{L}_{DLS-OPC} = \mathcal{L}_{DMO} + \mathcal{L}_{DLS} + \lambda_2 \mathcal{L}_{L_1}(\hat{y}, w_r). \quad (4)$$



(a)



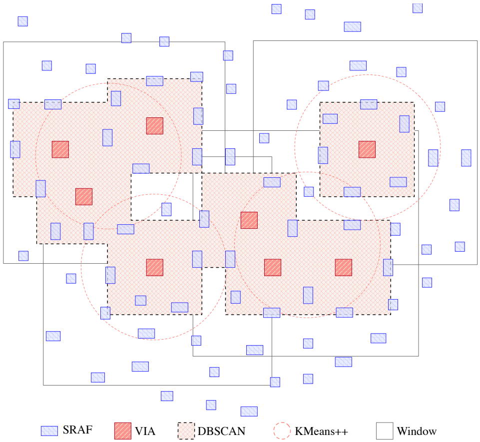
(b)

Irregular Splitting Algo: Coarse to Fine, DBSCAN to KMenas

Algo. detail

1. DBSCAN then KMeans++
2. Initialize the number of centroids from 1 to V to run KMeans++.
3. Every cluster contains no more than K via patterns.
4. Every via pattern must be contained in a window.
5. If (3) or (4) is not satisfied, increase the centroid number .

Algo. figure

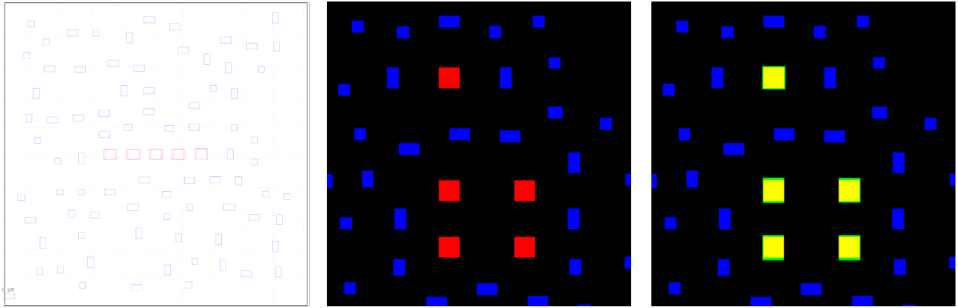


Main Contribution

- DCGAN-HD: we extend cGANs model by redesign the generator and discriminator for high resolution input (1024*1024), combined with a novel window-splitting algorithm, our model can handle input layout of any size with high accuracy.
- We build up a deep lithography simulator (DLS) based on our DCGAN-HD. Thanks to the express power of stack convolution layers, DLS is expected to conduct lithography simulation faster with similar contour quality compared to legacy lithography simulation process.
- We present DLS-DMO, a unified end-to-end trainable OPC engine that employs both DLS and DMO to conduct mask optimization without further fine-tune with legacy OPC engines.
- Experimental results show that the proposed DLS-OPC framework is able to output high quality lithography contours more efficiently than Calibre, which also derives $\sim 4\times$ speed-up in OPC tasks while generating masks with even better printability.

Outline

Results on self-generated datasets



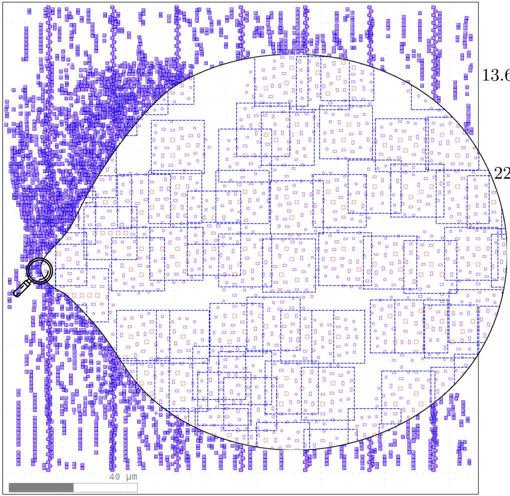
Layouts Design Mask

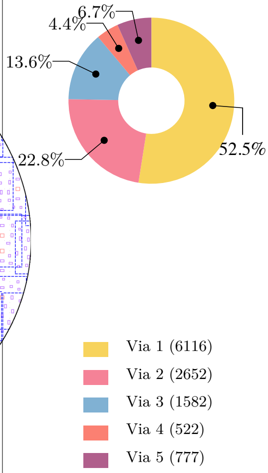
Results on self-generated datasets

Table 3: Comparison with State-of-the-art

	GAN-OPC				Calibre				DLS-OPC			
	EPE (nm)	L_2 (nm)	PVB (nm^2)	t(s)	EPE (nm)	L_2 (nm)	PVB (nm^2)	t(s)	EPE (nm)	L_2 (nm)	PVB (nm^2)	t(s)
via 1	1.9963	1464	3064	284	0.2270	1084	2918	1417	0.2760	**1080**	**2917**	321
via 2	4.0936	4447	5764	281	0.4143	2161	5595	1406	0.8245	**2129**	**5576**	336
via 3	4.7920	8171	8426	285	0.5415	3350	8286	1435	0.8213	**3244**	**8271**	317
via 4	4.8181	11659	11558	291	0.4951	4331	10975	1477	0.8777	**4263**	**10946**	327
via 5	5.0121	15773	13876	279	0.4851	5410	13663	1423	0.7341	**5396**	**13640**	318
via 6	5.0814	18904	16371	284	0.4947	6647	15572	1419	0.8472	**5981**	**15543**	320
Average	4.2989	10069	9843	284	0.4430	3831	9502	1430	0.7301	**3682**	**9482**	323
Ratio	5.888	2.735	1.038	0.879	0.607	1.040	1.002	4.427	1.00	**1.00**	**1.00**	1.00

Results on ISPD 2019 datasets





Results on ISPD 2019 datasets

Table 4: Results on ISPD 2019

	Calibre				DLS-OPC			
	EPE (nm)	L_2 (nm)	PV Band (nm^2)	Runtime (s)	EPE (nm)	L_2 (nm)	PV Band (nm^2)	Runtime (s)
via 1(6116)	0.2728	1073	2857	18959	0.3196	**1056**	**2848**	**3963**
via 2(2652)	0.4548	2232	5670	7537	0.8015	**2172**	**5654**	**1742**
via 3(1582)	0.4980	3602	8276	4494	1.1472	**3196**	**8127**	**1021**
via 4(522)	0.5336	4395	11051	1692	1.1967	**4361**	**10946**	**341**
via 5(777)	0.8214	5526	12305	2537	1.2696	**4542**	**12251**	**495**
Weighted Average	0.393	2126	5230	12525	0.644	**1981**	**5193**	**2664**
Ratio	0.615	1.073	1.007	4.70	1.00	**1.00**	**1.00**	**1.00**

Results on ISPD 2019 datasets

Table 5: Results on ISPD 2019 Large layout

	EPE (nm)	PVB (um^2)	Runtime(s)		
			Prep.	Infer.	Total
DLS-OPC	0.7828	61.3132	4395	231.5	4626.5
Calibre	0.5372	65.7928	6111.46		

Results

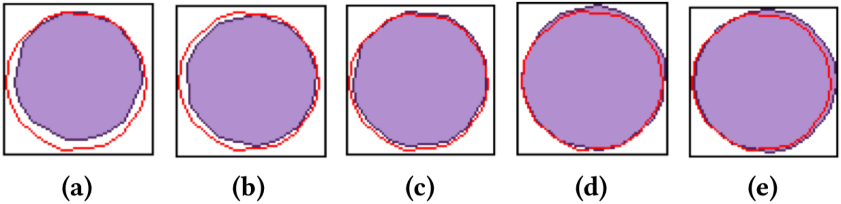


Figure 8: Visualization of DLS-OPC model advancement on via layer: (a) Epoch 20; (b) Epoch 40; (c) Epoch 60; (d) Epoch 80; (e) Epoch 100.

Thanks

Thank you.