Layout Pattern Generation and Legalization with Generative Learning Models

Xiaopeng Zhang[1], James Shiely[2], Evangeline F.Y. Young[1]

[1]The Chinese University of Hong Kong

[2]Synopsys Inc.

Biography

- Xiaopeng Zhang
- The Chinese University of Hong Kong
- xpzhang@cse.cuhk.edu.hk

- He is a postgraduate student studying for his Ph.D. degree in the Department of Computer Science and Engineering, The Chinese University of Hong Kong, under the supervision of Prof. Evangeline F.Y. Young.

- His current research interest includes machine learning in physical design and design for manufacturability.

Outline

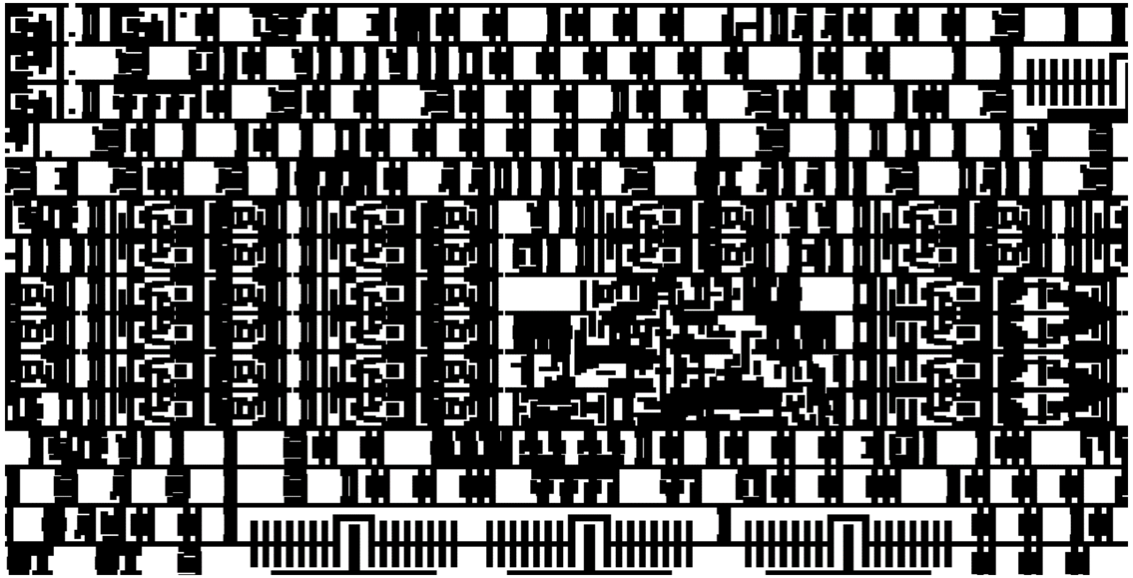
- Background & Introduction
- Algorithms
- Experimental Results
- Conclusion

Outline

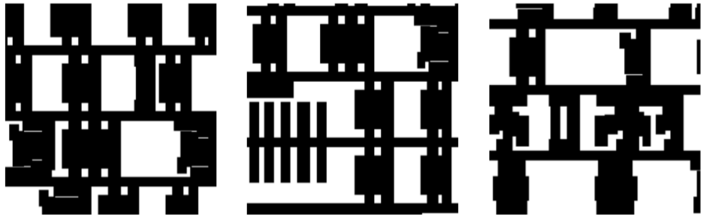
- Background & Introduction
- Algorithms
- Experimental Results
- Conclusion

Application of Pattern Generation

- Hotspot detection and fix
- Working in early technology node development
- Training large deep learning models
- ……



Parts of an example design layout [1]



Examples of layout patterns.

[1] Topaloglu, Rasit O. "ICCAD-2014 CAD contest in design for manufacturability flow for advanced semiconductor nodes and benchmark suite." 2014 IEEE/ACM ICCAD, 2014.

Related Works

- Generating new patterns by rotation, flipping and moving edges on existing patterns (limited productivity and diversity)
- Guiding a random generator to explore new territories of the design space (complex manual guidance, limited diversity)
- Learning-based techniques (violating design rules, hardly working in complex two-dimensional layouts)
- Previous works didn’t consider the style of generated patterns.

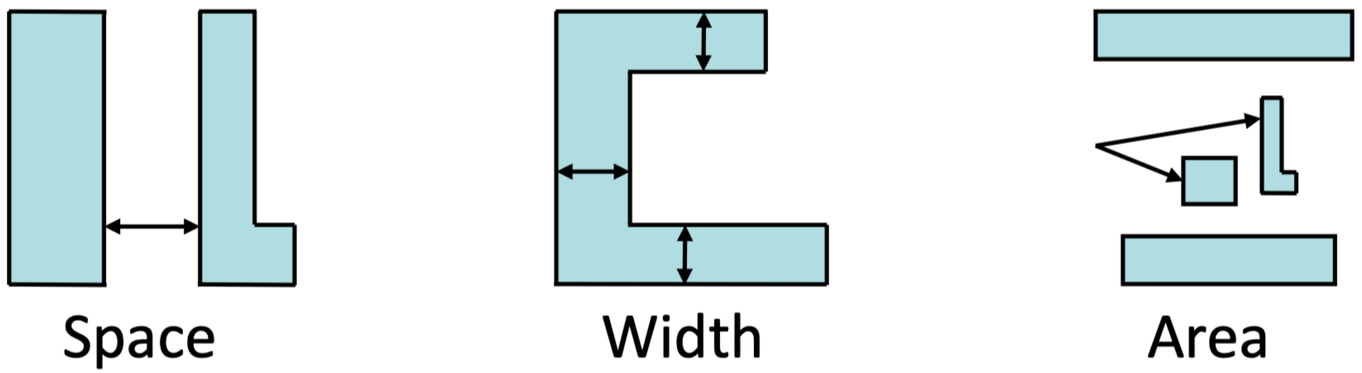
Our task

- Design a generative learning-based pattern generation framework which can generate realistic two-dimensional patterns with little manual guidance.

- Develop a learning-based legalization tool to fix the generated patterns with DRC violations.

- Build a model to check pattern styles and filter out unrealistic generated patterns.

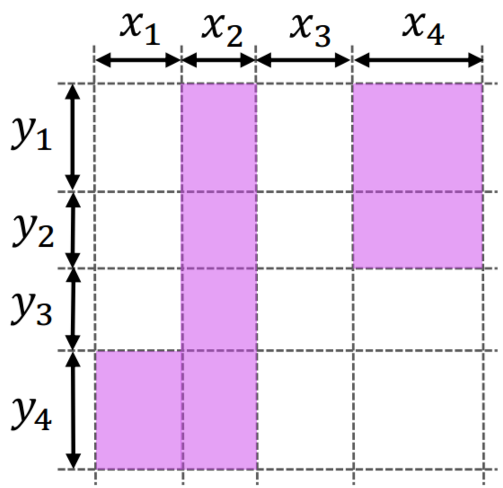
Satisfying Design Rules



- ***Space*** represents the distance between two adjacent polygons.
- ***Width*** measures the shape size in one direction.
- ***Area*** denotes the area of a polygon.

A layout pattern is legal if and only if all these measurements are not less than some given thresholds.

Squish pattern representation



(a) Splitting original pattern.

$$P_T = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

$$\delta_x = [x_1 \quad x_2 \quad x_3 \quad x_4]$$

$$\delta_y = [y_1 \quad y_2 \quad y_3 \quad y_4]$$

(b) Squish topology and geometry.

- storage-efficient (2048*2048 -> 128*128)
- lossless
- conducive to convolutional neural network models

Evaluation Metric

For checking the quality of the generated patterns:

Definition 1. *(Pattern Validity). The pattern validity, denoted as V, is the proportion of realistic patterns to the total patterns, as shown in Equation* (1),

$$V = \frac{The\ number\ of\ realistic\ patterns}{The\ number\ of\ total\ patterns}. \tag{1}$$

For checking if the generated patterns are diverse :

Definition 2. *(Pattern Diversity). The pattern diversity, denoted as H, is the Shannon entropy of the pattern complexity sampled from the pattern library, as shown in Equation* (2),

$$H = -\sum_i \sum_j P(c_{xi}, c_{yj}) \log P(c_{xi}, c_{yj}), \tag{2}$$

where $P(c_{xi}, c_{yj})$ *is the probability of a pattern with complexities of* c_{xi} *and* c_{yj} *sampled from the pattern library.*

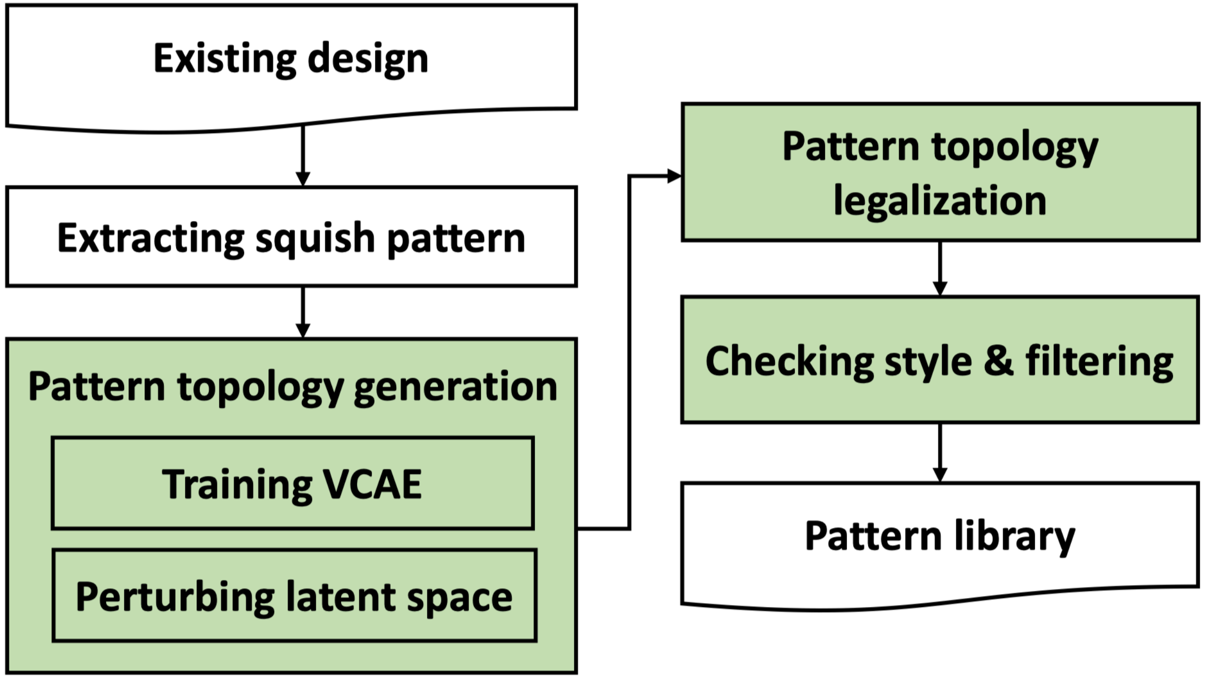
Problem Statement

Problem 1. *(Pattern Generation) Given a set of actual IC layout patterns and design rules, the objective of pattern generation is to generate a legal pattern library such that the pattern diversity and the number of unique realistic patterns in the library are maximized.*

Outline

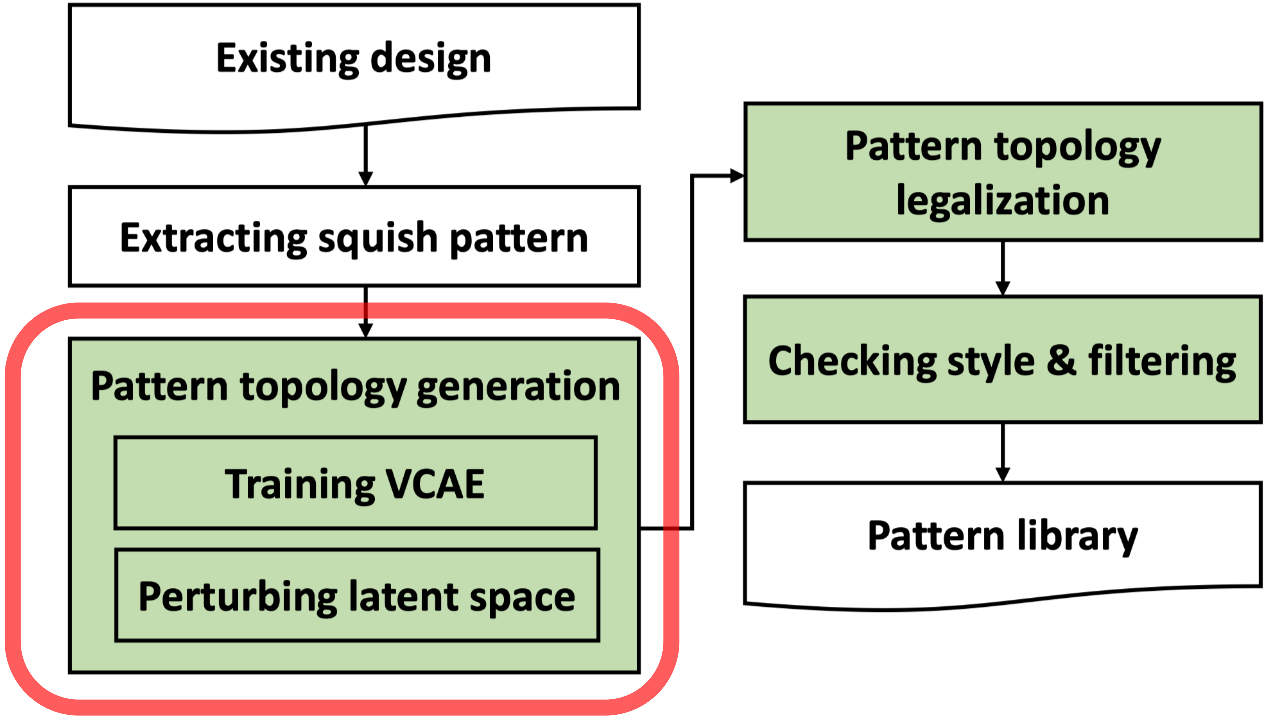
- Background & Introduction
- Algorithms
- Experimental Results
- Conclusion

Overview



The overall flow of *CU p*attern generation and legalization framework (CUP)

Pattern Topology Generation

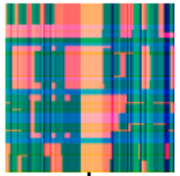


The overall flow of *CU p*attern generation and legalization framework (CUP)

Pattern Topology Generation

Goal:

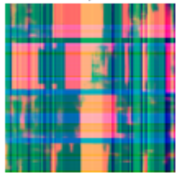
Topology P



Topology generation model

Random perturbation $\leftarrow \Delta\ell$

New topology P’



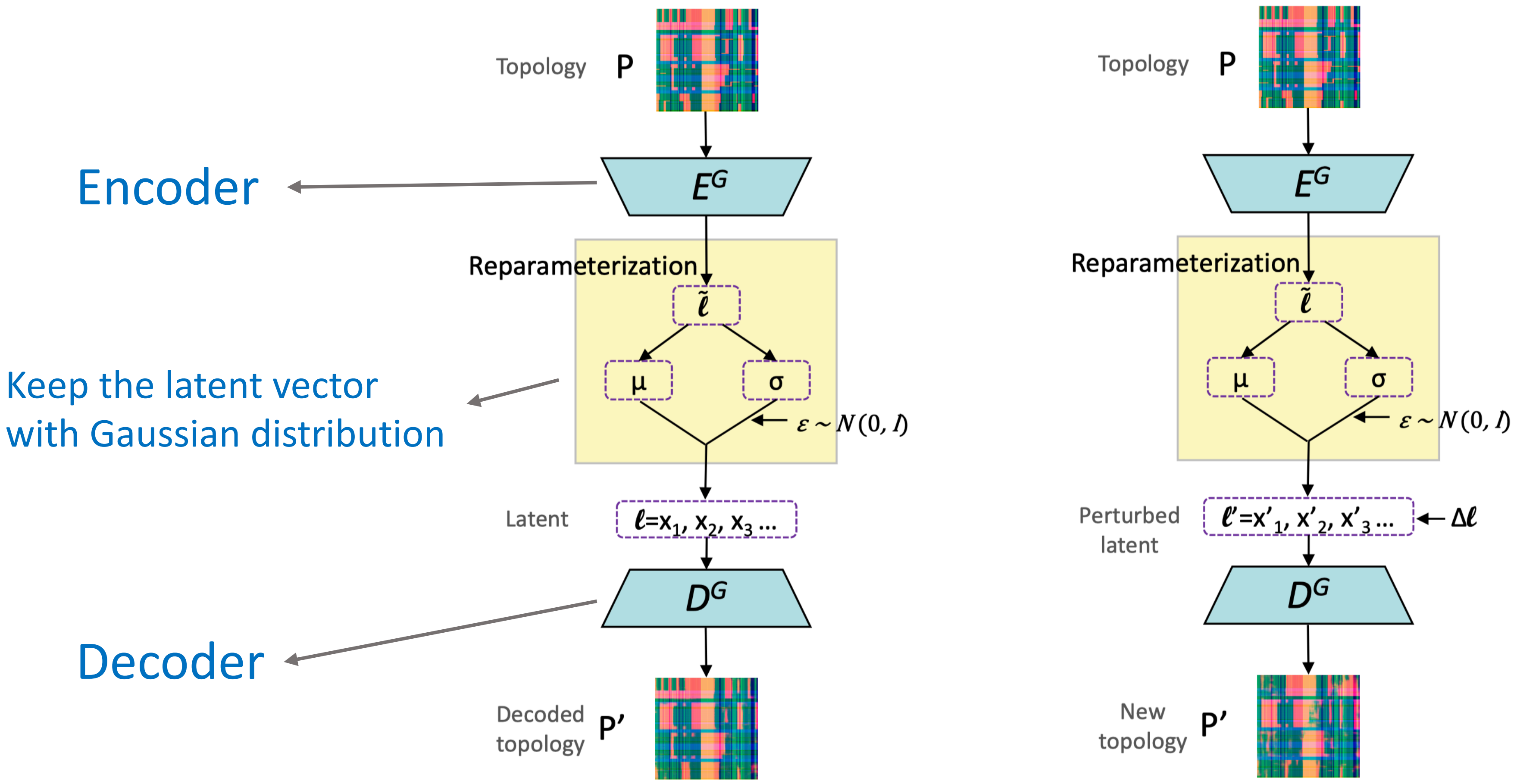
$$P = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

$$\delta_x = [x_1 \quad x_2 \quad x_3 \quad x_4]$$

$$\delta_y = [y_1 \quad y_2 \quad y_3 \quad y_4]$$

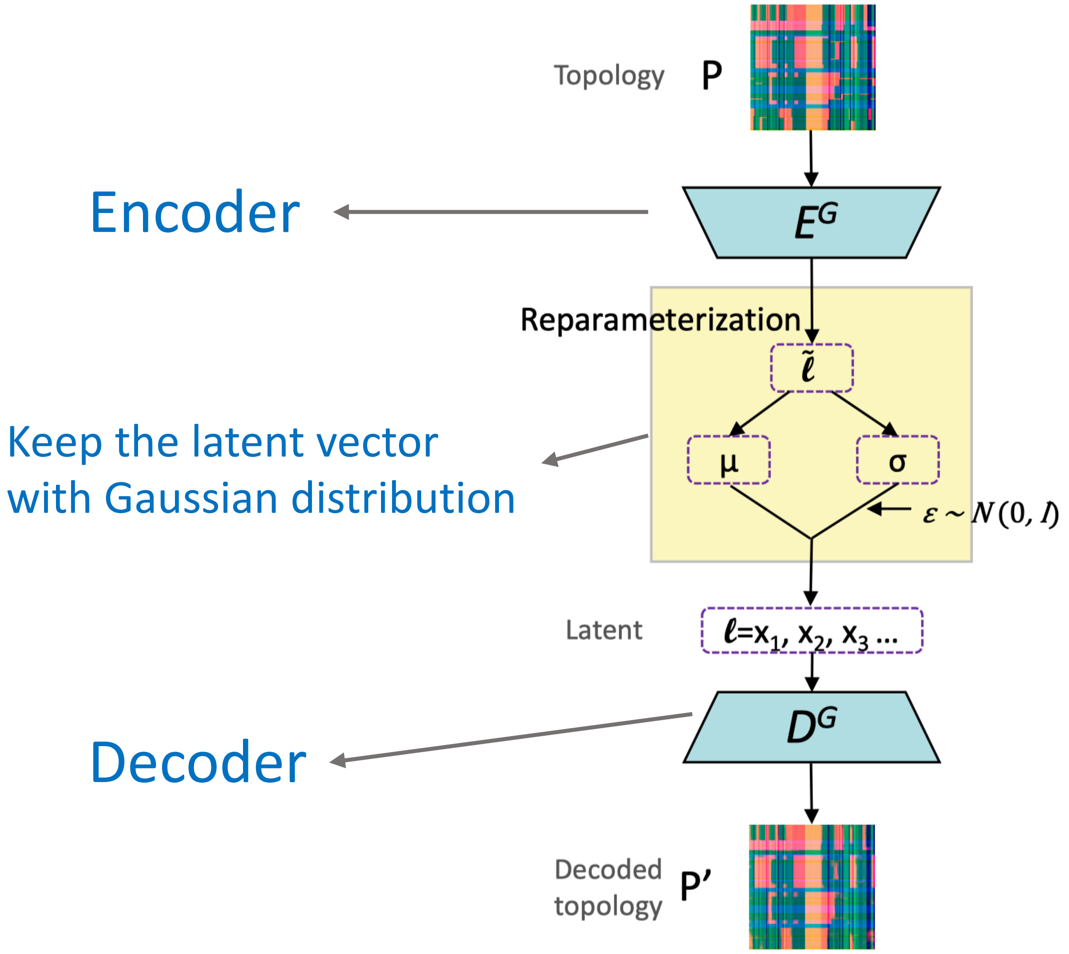
Topology in squish pattern representation

Variational Convolutional Auto-Encoder (VCAE)



(a) Training. **(b) Testing.**

Variational Convolutional Auto-Encoder (VCAE)

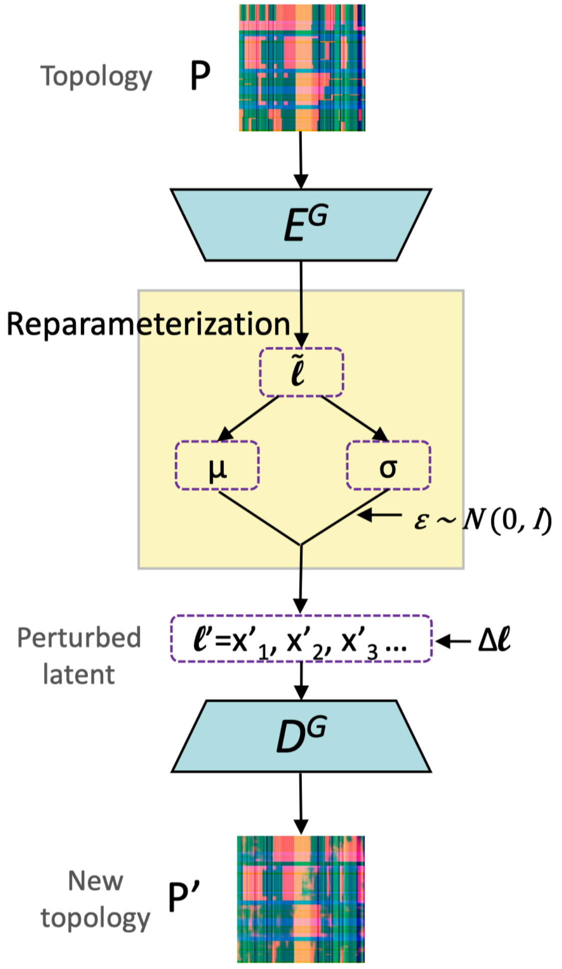


Kullback-Leibler (KL) divergence

Objective function:

$$
\begin{aligned}
min \quad & \boxed{D_{KL}(\mathcal{N}(\mu, \sigma^2) \| \mathcal{N}(0, I))} + \lambda \|P - P'\|_F^2, \\
s.t. \quad & \tilde{l} = E^G(P, W_E), \\
& \mu = f_R^{\mu}(\tilde{l}, W_R^{\mu}), \\
& \sigma = f_R^{\sigma}(\tilde{l}, W_R^{\sigma}), \\
& l = \mu + \sigma \odot \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, I), \\
& P' = D^G(l, W_D).
\end{aligned}
$$

Generating New Topology



Generate new pattern topology
by random Gaussian perturbation Δl :

$$P' = D^G(l + \Delta l, W_D),$$

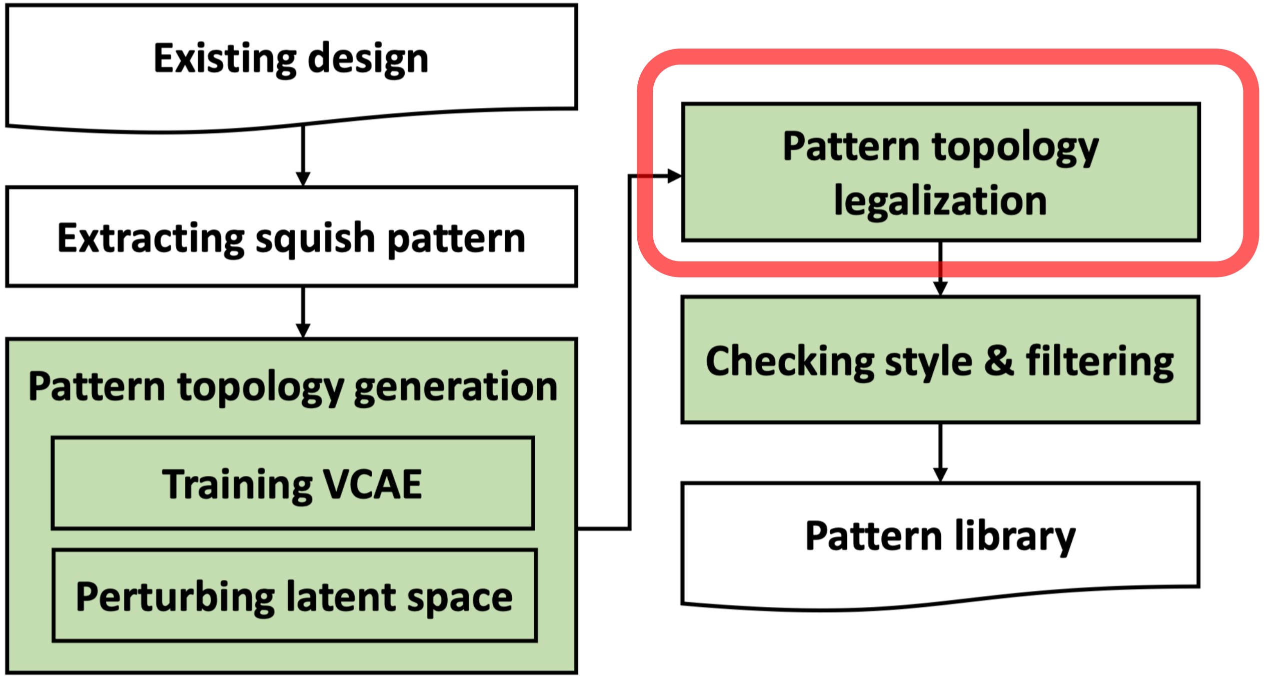
$$\text{s.t.} \quad \Delta l_i \sim \mathcal{N}(0, c^2), \quad i = 1 \cdots N$$

Network Configuration of VCAE

Encoder		Decoder	
Layer	Output Size	Layer	Output Size
Input	$1 \times 128 \times 128$	Latent	$4 \times 16 \times 16$
Conv-IN-ReLU	$64 \times 128 \times 128$	Deconv-IN-ReLU	$32 \times 16 \times 16$
Conv-IN-ReLU	$128 \times 64 \times 64$	Deconv-IN-ReLU	$64 \times 16 \times 16$
Conv-IN-ReLU	$256 \times 32 \times 32$	Deconv-IN-ReLU	$128 \times 16 \times 16$
ResBlock	$256 \times 32 \times 32$	ResBlock	$128 \times 16 \times 16$
Conv-IN-ReLU	$256 \times 16 \times 16$	ResBlock	$128 \times 16 \times 16$
Conv-IN-ReLU	$128 \times 16 \times 16$	Deconv-IN-ReLU	$256 \times 16 \times 16$
ResBlock	$128 \times 16 \times 16$	Deconv-IN-ReLU	$256 \times 32 \times 32$
ResBlock	$128 \times 16 \times 16$	ResBlock	$256 \times 32 \times 32$
Conv-IN-ReLU	$64 \times 16 \times 16$	Deconv-IN-ReLU	$128 \times 64 \times 64$
Conv-IN-ReLU	$32 \times 16 \times 16$	Deconv-IN-ReLU	$64 \times 128 \times 128$
Conv-IN-ReLU	$4 \times 16 \times 16$	Deconv-Tanh	$1 \times 128 \times 128$

Reparameterization for μ		**Reparameterization** for σ	
Layer	Output Size	Layer	Output Size
Input	1024	Input	1024
Linear	1024	Linear	1024

Pattern Topology Legalization

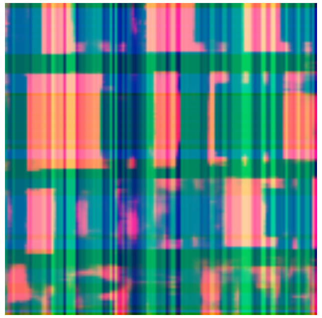


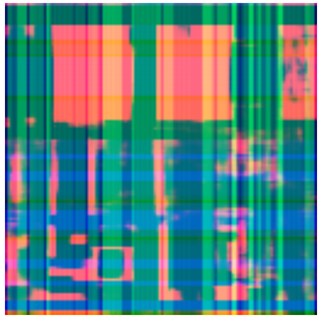
The overall flow of *CU p*attern generation and legalization framework (CUP)

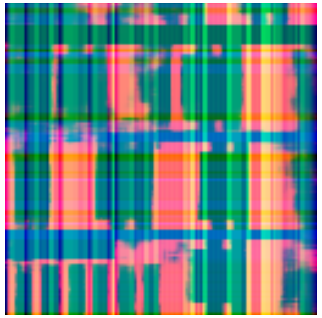
Generated Topologies from VCAE

Generated topologies:

Blurry flaws!

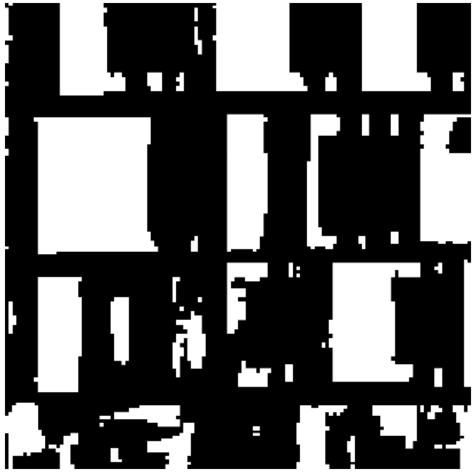






Generated layout patterns:

DRC violations!



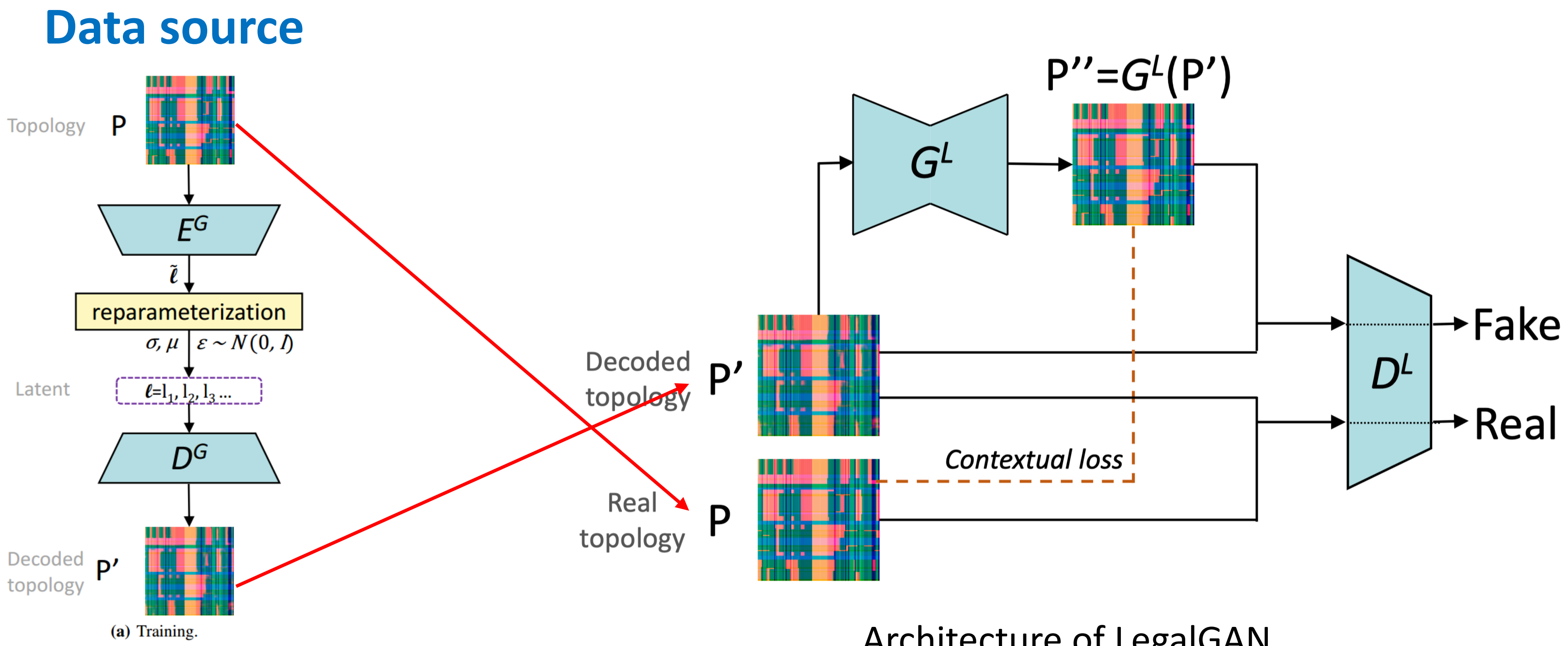
Because VAE often suffers from blurry generation due to the injected noise and imperfect element-wise measures.

Pattern Topology Legalization

- Motivation: eliminate blurry flaws and DRC violation risks by reducing the reconstruction loss from VCAE.
- Method: LegalGAN, a generative model based on CGAN

LegalGAN

LegalGAN: legalization model based on CGAN



Architecture of LegalGAN

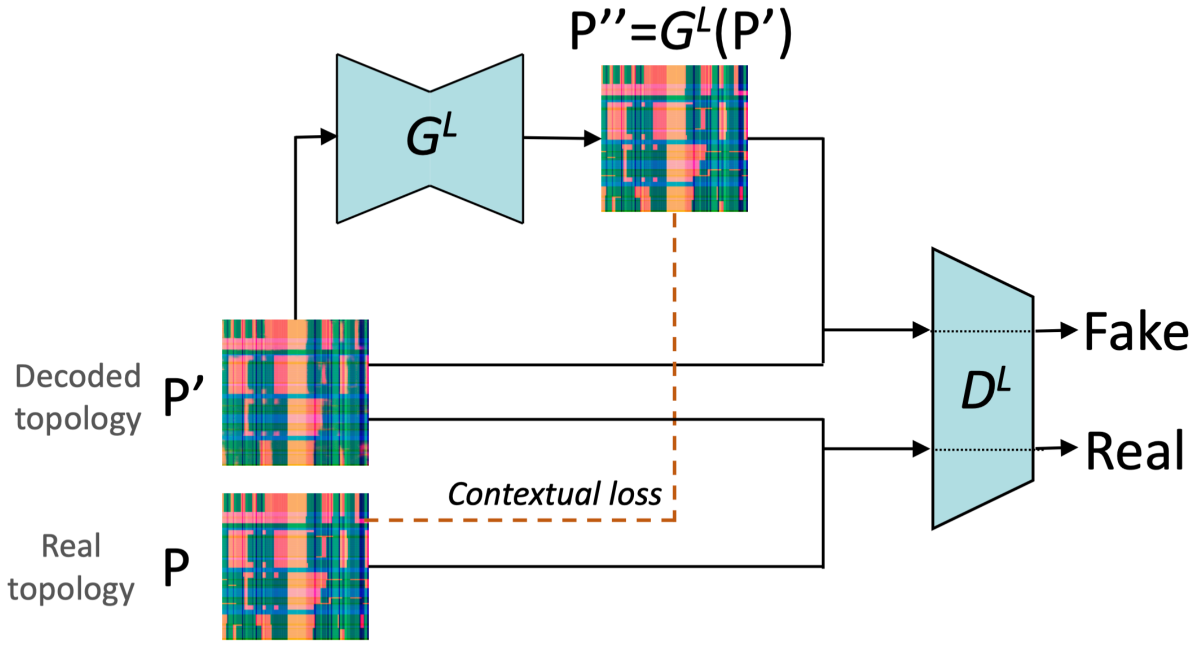
Basic idea: eliminate blurry flaws and DRC violation risks by reducing the reconstruction loss from VCAE.

LegalGAN

LegalGAN: legalization model based on CGAN

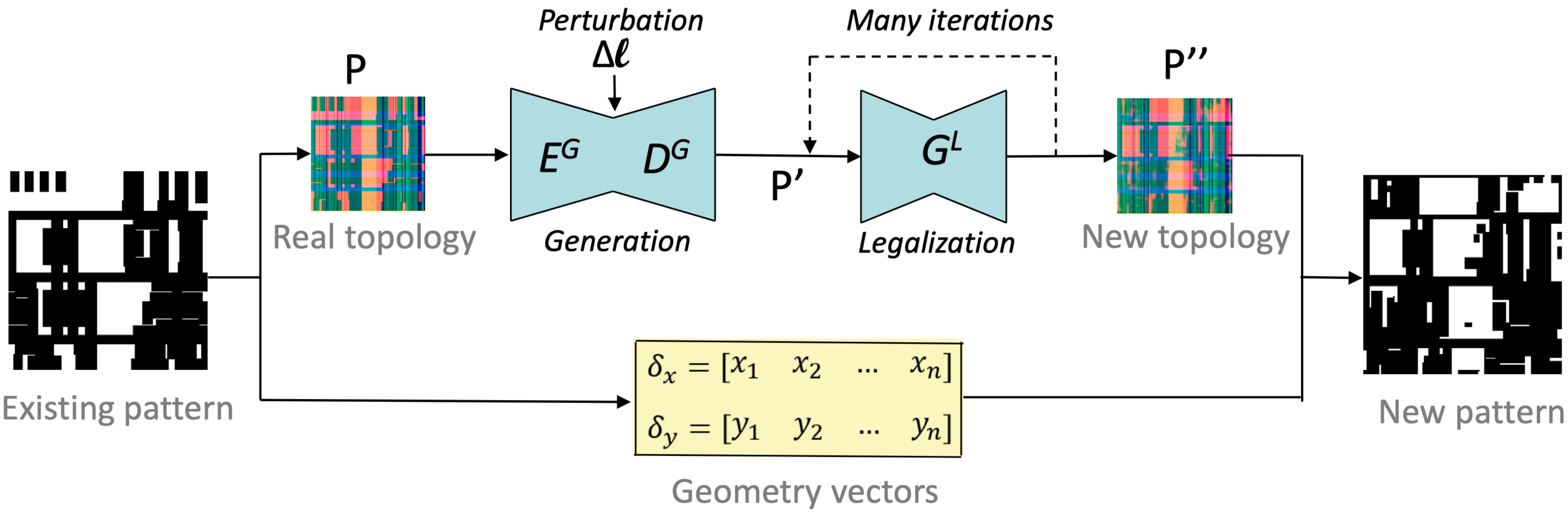
Objective function:

$$\min_{G^L} \max_{D^L} \mathbb{E}_{P,P'}\left[\log(D^L(P',P))\right] + \mathbb{E}_{P'}\left[\log(1 - D^L(P', G^L(P')))\right] + \gamma \cdot \mathbb{E}_{P'}\left[||P - G^L(P')||_1\right]$$



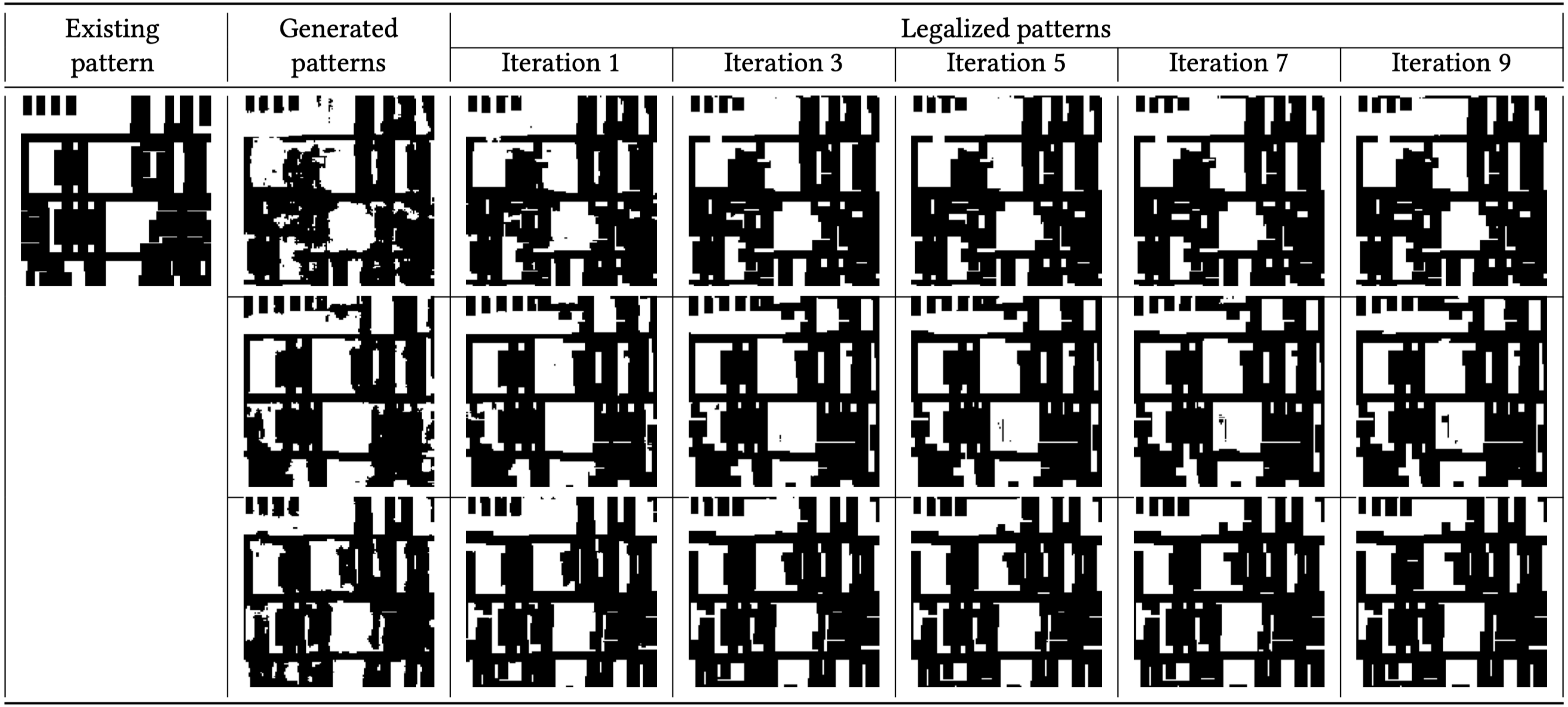
Architecture of LegalGAN

Overall Flow in Testing Stage

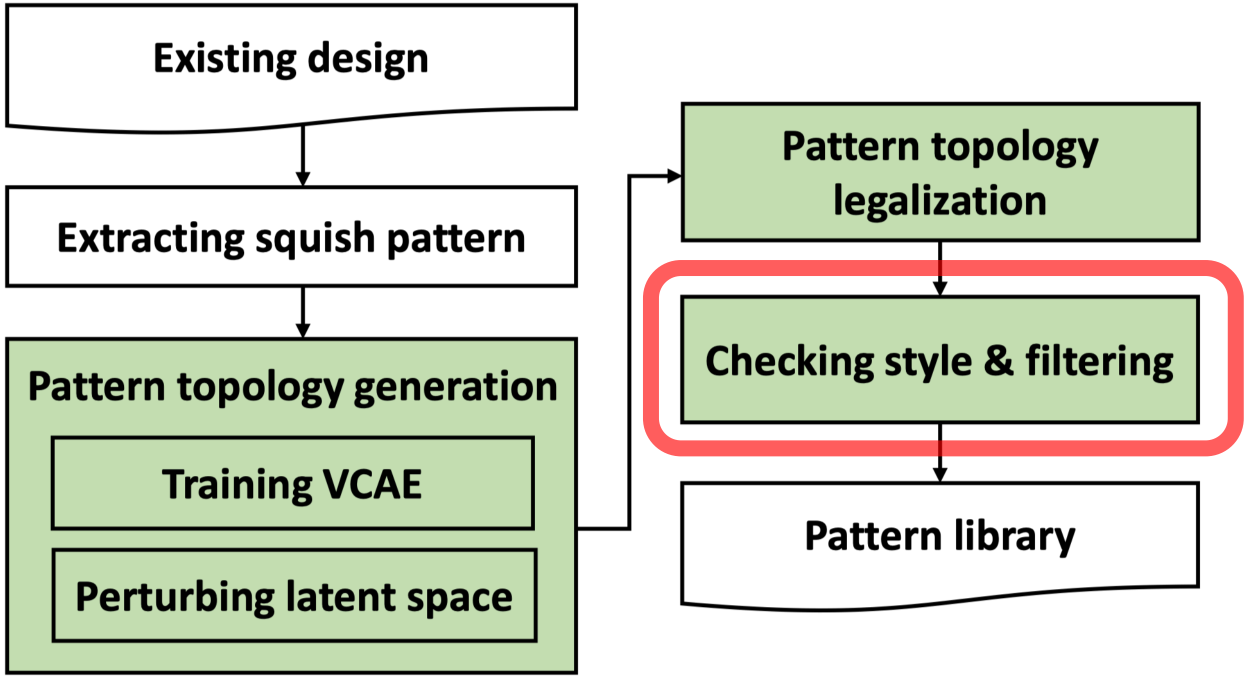


The flow of pattern topology generation and legalization in the testing stage.

Visualization of the LegalGAN process



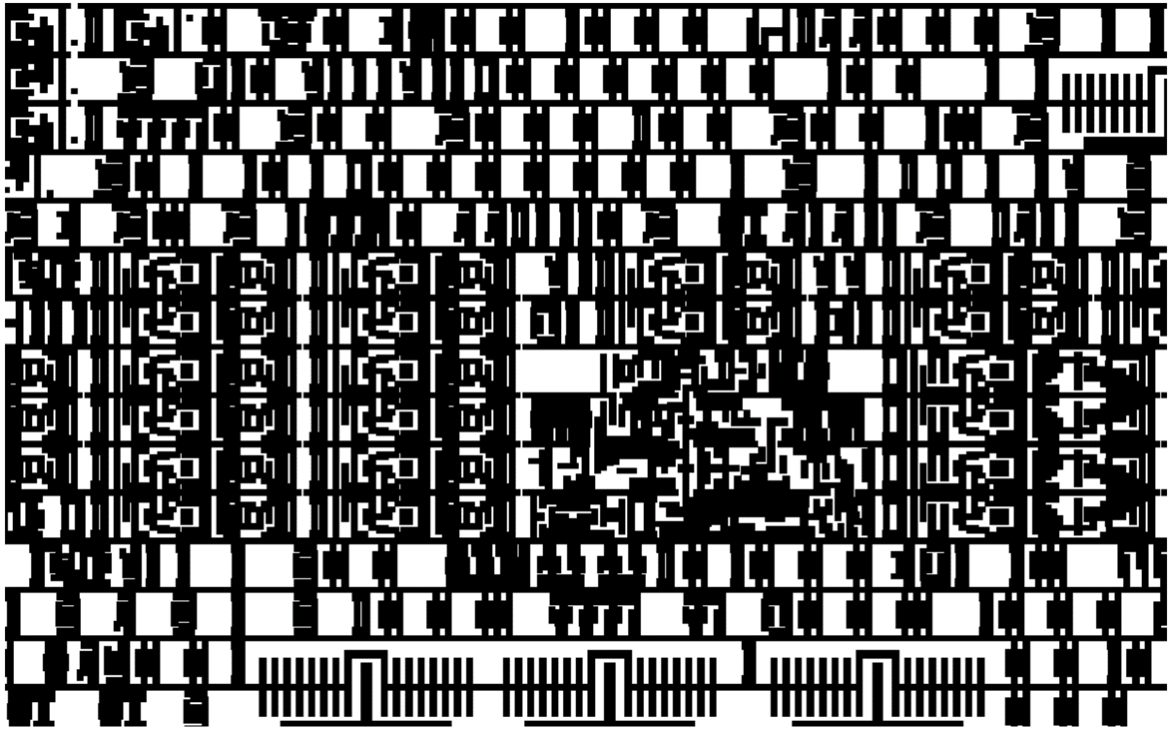
Pattern Style Detection



The overall flow of *CU p*attern generation and legalization framework (CUP)

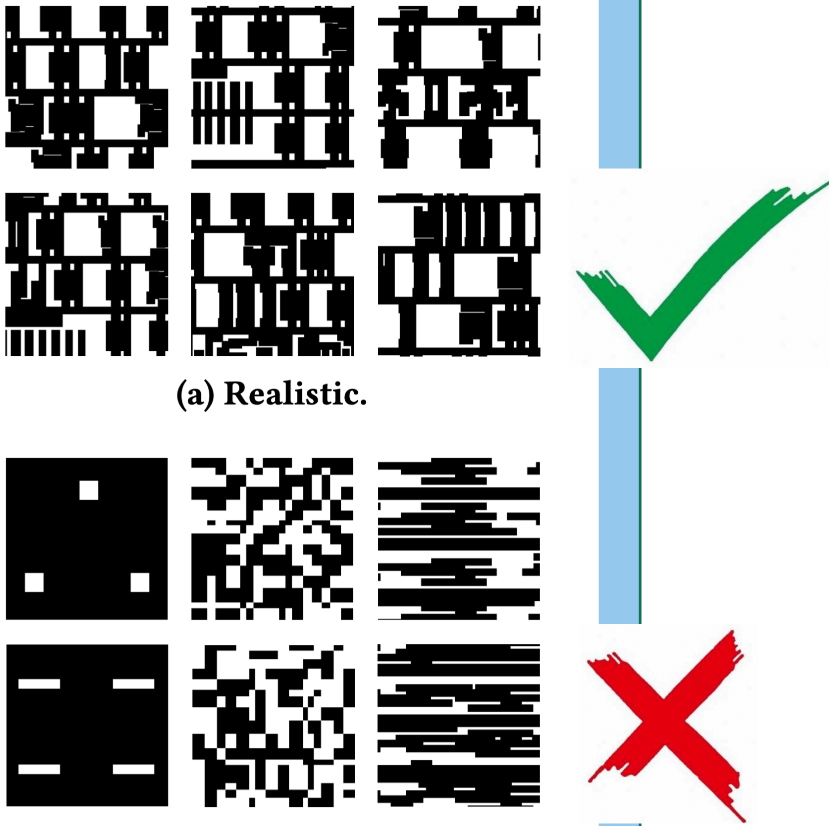
Pattern Style: Realistic and Unrealistic

A question: Must a DRC-clean pattern be a realistic pattern?



Parts of an actual IC layout

New patterns:



(a) Realistic.

(b) Unrealistic.

Pattern Validity

Realistic pattern: a pattern that closely resembles the actual layout and shares the same inherent style with the layout.

For checking the quality of the generated patterns:

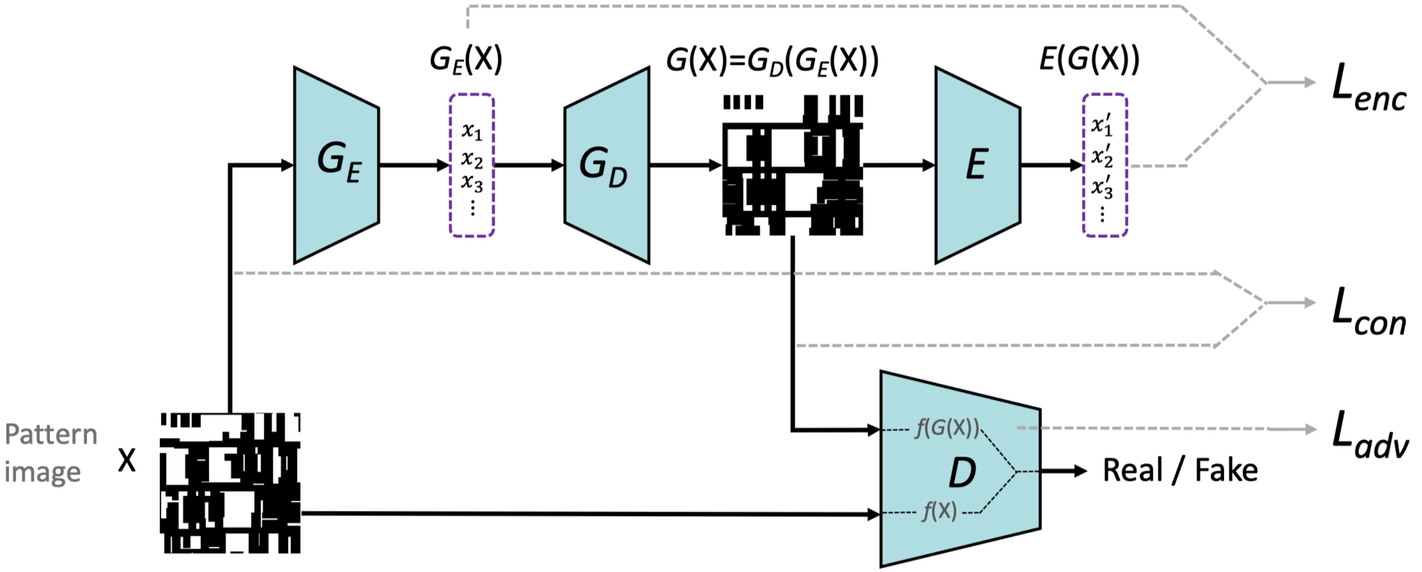
Definition 1. *(Pattern Validity). The pattern validity, denoted as V, is the proportion of realistic patterns to the total patterns, as shown in Equation (1),*

$$V = \frac{The\ number\ of\ realistic\ patterns}{The\ number\ of\ total\ patterns}. \tag{1}$$

How to detect the pattern style?

Basic idea: *anomaly detection based on CGAN.*

Pattern Style Detection



Architecture of pattern style detection model.

$$L_{enc} = \mathbb{E}_X\left[||G_E(X) - E(G(X))||_2\right]$$

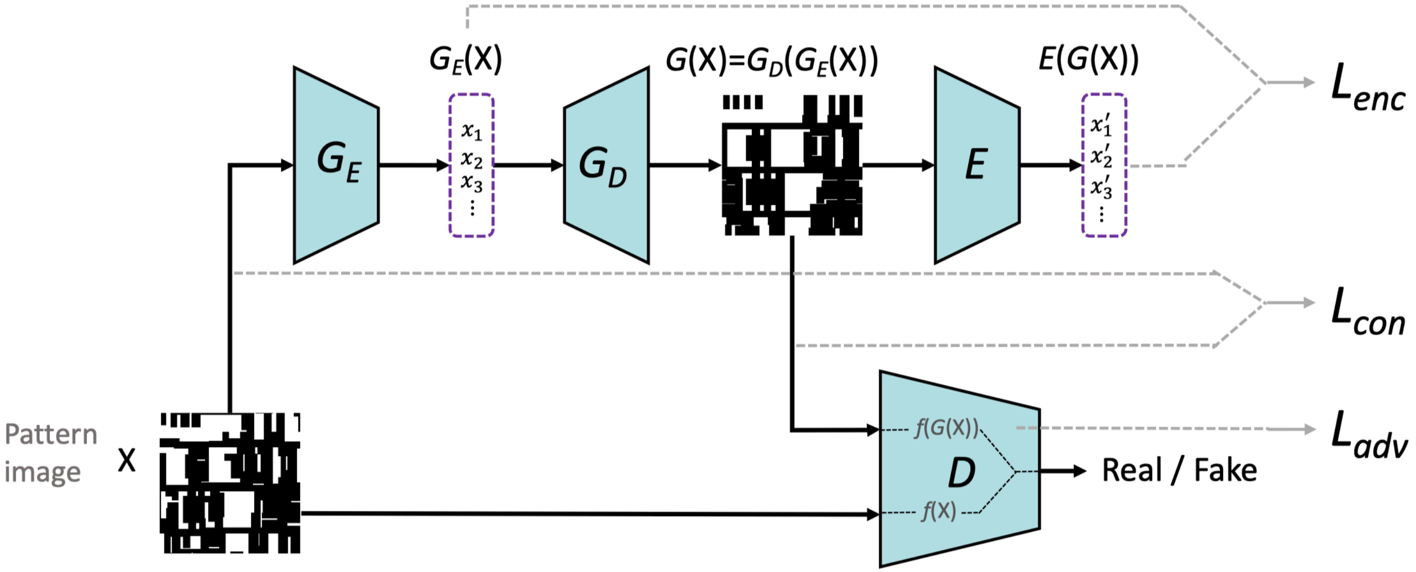
$$L_{con} = \mathbb{E}_X\left[||X - G(X)||_1\right]$$

$$L_{adv} = \mathbb{E}_X\left[||f(X) - f(G(X))||_2\right]$$

Objective function:

$$\min_{G_E,G_D,E} \max_{D} L_{adv} + \alpha \cdot L_{con} + \beta \cdot L_{enc}$$

Pattern Style Detection



Architecture of pattern style detection model.

For a new pattern,
the pattern anomaly score:

$$score(\hat{x}) = ||G_E(\hat{x}) - E(G(\hat{x}))||_1$$

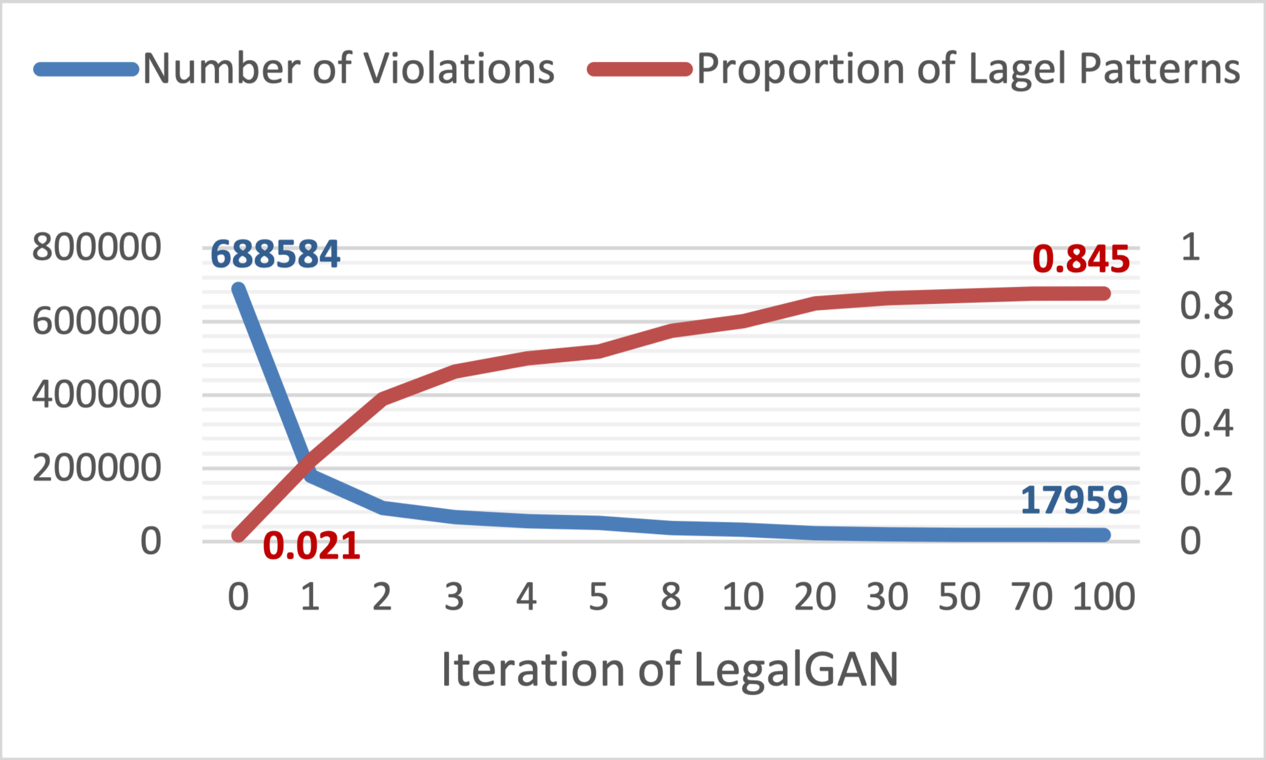
The pattern is realistic if:

$$score(\hat{x}) < T$$

Outline

- Background & Introduction
- Algorithms
- Experimental Results
- Conclusion

Effectiveness of LegalGAN



- **97.39%** DRC violations are removed (from 688584 to 17959).
- The proportion of legal patterns in the total changes **from 2.1% to 84.5%**

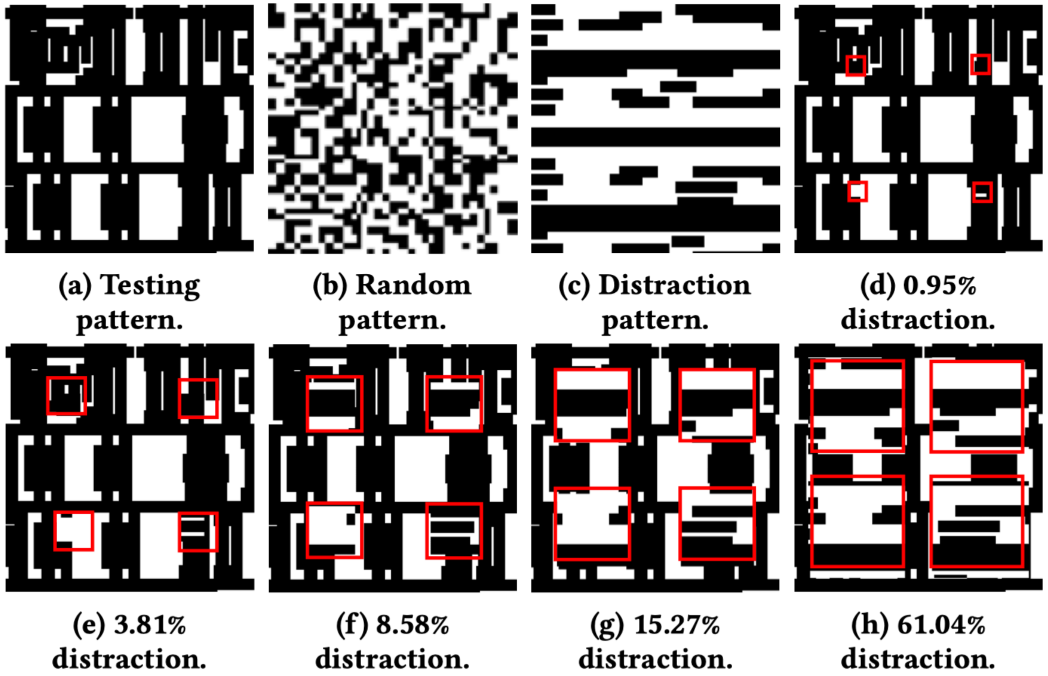
Results on Pattern Diversity

Set/Method	All patterns		Legal patterns	
	Pattern #	Diversity (H)	Pattern #	Diversity (H)
Existing patterns	13869	10.7767	-	-
Basic patterns	100	6.6039	-	-
CAE	100000	4.5875	19	3.7871
VCAE	100000	10.9311	2126	9.9775
CAE+LegalGAN	100000	5.8465	3740	5.8142
CUP (VCAE+LegalGAN)	100000	9.8692	84510	9.8669

[1] Pattern# means the number of patterns.

- CUP generate the most new patterns with high pattern diversity H = 9.8669

Validation of Style Detection Model



Pattern Set	Pattern #	Pattern Validity (V)		
		$T = 0.6$	$T = 0.7$	$T = 0.8$
Testing patterns	3000	0.6493	0.8485	0.9306
Random patterns	3000	0	0	0.007
Distraction patterns	3000	0.0119	0.0143	0.0246
0.95% distraction	3000	0.5981	0.8236	0.9198
3.81% distraction	3000	0.4227	0.6943	0.8536
8.58% distraction	3000	0.0710	0.2204	0.4371
15.27% distraction	3000	0.0058	0.0365	0.1041
61.04% distraction	3000	0.0003	0.0041	0.0191

- The sets of more realistic patterns can get larger pattern validity score, which means the trained pattern style detection model has the ability of checking the layout style.

Results on Pattern Validity

Set/Method	Pattern #	Pattern Validity (V)		
		$T = 0.6$	$T = 0.7$	$T = 0.8$
Testing patterns	3000	0.6493	0.8485	0.9306
CAE+LegalGAN	2126	0.0003	0.0027	0.0167
CUP (VCAE+LegalGAN)	84510	0.5430	0.7840	0.9057

- CUP can generate a pattern library with high pattern validity.

Outline

- Background & Introduction
- Algorithms
- Experimental Results
- Conclusion

Conclusion

- We propose a novel two-stage generative learning-based pattern generation framework including pattern topology generation and legalization, which has the capability of generating realistic patterns with little manual guidance.

- A pattern style detection tool is designed to check pattern styles and filter out unrealistic generated patterns.

- Experimental results demonstrate that our framework has the capability of generating realistic legal patterns with high pattern diversity.

THANKS