TopoCellGen: Generating Histopathology Cell Topology with a Diffusion Model
— Supplementary Material —

In the supplementary material, we begin with notations for foreground and background in Sec. 6, followed by a description of the background knowledge about persistent homology in Sec. 7. Next, we provide detailed introduction to our layout-guided pathology image generation part in Sec. 8, followed by the comprehensive descriptions of the datasets in Sec. 9. The implementation details are provided in Sec. 10. In Sec. 11, we discuss the evaluation metrics in detail. To ensure the generation accuracy, we conduct the analysis on cell count distribution across training and test sets in Sec. 12. More ablation studies are given in Sec. 13. The biological plausibility analysis by the domain expert is provided in Sec. 14. Then, we provide the spatial point pattern analysis using multivariate Ripley's K-functions in Sec. 15, followed by the discussion on computational cost and scalability of our method in Sec. 16. Finally, we discuss the limitations of our method in Sec. 17.

6. Notes on Foreground and Background

Here, we provide some notations about foreground and background in our paper. In our experiments, the background of the layouts is black (the pixel value of 0) as can be seen in Fig. 2 and Fig. 5. For better visualization, we display the multi-class cell layouts with white as the background in Fig. 1, Fig. 3 and Fig. 4 of the main paper.

7. Background: Persistent Homology

In algebraic topology [49], homology classes capture topological features across different dimensions. For instance, 0-, 1-, and 2-dimensional structures represent connected components, loops (or holes), and voids, respectively. In binary images, the number of d-dimensional topological features is described by the d-dimensional Betti number, β_d.[1] While topological structures are well-defined in binary images, extending this theory to real-world data, which is often continuous and noisy, poses challenges.

In the case of analyzing cell point clouds, where data is inherently discrete, we require a robust framework to infer the underlying topological structures. Persistent homology, developed in the early 2000s [13, 14], addresses this need by tracking the evolution of topological features across multiple scales.

Given a point cloud in the 2D space $P \subseteq \mathbb{R}^2$, a filtration is built by considering a growing family of simplicial complexes constructed from the point cloud as a function of a parameter (e.g., radius). For each parameter value, we define a set of simplices connecting the points, starting from isolated vertices and gradually adding edges and higher-dimensional simplices as the parameter increases. This creates a series of nested simplicial complexes: $\varnothing \subseteq K_{r_1} \subseteq K_{r_2} \subseteq ... \subseteq K_{r_n}$. As the parameter grows, the topology of the complexes changes, with new connected components and loops emerging or vanishing.

Persistent homology captures these changes, tracking the birth and death of topological features over the filtration. The result is summarized in a persistence diagram (Dgm), which provides a multi-scale representation of the topological structures. A Dgm consists of points in a 2D plane, each representing a topological feature. The coordinates of each point, (b, d), correspond to the feature's birth and death filtration values, providing a concise description of its persistence across scales.

[1] Technically, β_d measures the dimension of the d-dimensional homology group. The number of distinct homology classes is exponential in β_d.

8. Layout To Image Generation

In this section, we introduce our layout-guided image generation framework in detail. The framework leverages a guided diffusion model (ADM) [11] to generate H&E images conditioned on multi-class cell layouts. The layouts serve as explicit conditional inputs to the diffusion model, which learns to reconstruct high-resolution pathology images from noisy counterparts during the reverse diffusion process. The conditioning mechanism is implemented using a cross-attention layer that seamlessly integrates cell layout information into the diffusion model. As shown in Fig. 6, the generated H&E images generated by the model accurately depict the relative densities and arrangements of different cell types, while preserving the fine-grained details characteristic of histopathology images, such as nuclear shapes and staining patterns. This helps greatly improve the performance of downstream tasks, such as cell detection and classification.

9. Details of the Datasets

BRCA-M2C dataset [1] is obtained from the TCGA dataset and contains 80, 10, and 30 pathology image patches for training, validating, and testing, respectively. This dataset provides dot annotations for multi-class classification in breast cancer images. All images are around 500×500 pixels. The cell classes are lymphocytes, tumor or epithelial, and stromal cells.

Lizard Dataset [22] is a large-scale resource for nuclear instance segmentation and classification, specifically target-

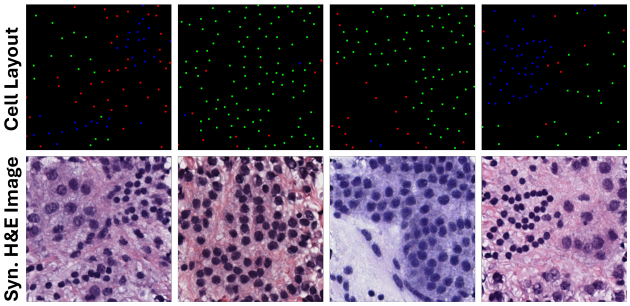

Figure 6. Qualitative results generated by our layout and image generation framework for downstream tasks.

ing colonic tissue in computational pathology. It includes nearly $495,000$ manually and semi-automatically annotated nuclei, categorized into six classes: epithelial cells, connective tissue cells, lymphocytes, plasma cells, neutrophils, and eosinophils. 238 images in the dataset are sourced from 6 publicly available datasets, ensuring diverse representations of normal, inflammatory, dysplastic, and cancerous colonic conditions.

10. Implementation Details

Our work is mainly based on guided-diffusion (ADM) [11]. The condition of our model is a list of cell counts. An embedding of the condition is obtained by using an encoding network. After that, we feed this embedding to all the residual blocks in the network by adding it to the timestep embedding [50]. For every dataset, the image resolution is 256×256. Our diffusion models use a cosine noise scheduler [50], with noising timesteps of 1000 for training. We first pre-train the diffusion model using only $\mathcal{L}_{simple}$ for 150K steps, then train with the three losses for 210K steps. During the inference, we use 100 steps of DDIM [62]. The learning rate is 2×10^{-5} and the batch size is 5. λ_c, λ_{intra} and λ_{inter} are all set to 0.0005.

For the layout-guided generation model, the learning rate is also 2×10^{-5} and we train the model only using $\mathcal{L}_{simple}$ for 360K steps. The batch size is 6. The image resolution is also 256×256. These experiments were conducted on 1 NVIDIA RTX A6000 GPU with 48GB RAM.

Our experiments designate specific test sets for each dataset to evaluate the synthetic cell layout generation process. For the BRCA-M2C dataset, we utilize 30 images in the test set, which were pre-defined in the dataset. To prepare these for testing, each image is segmented into patches using a sliding window approach with a stride of 32 pixels, resulting in patches of size 256×256. This process yields a total of $1,550$ patches for the BRCA-M2C dataset. We randomly select 20% of the cell layouts as the test set for the Lizard dataset, which lacks predefined training and test splits. The chosen images undergo the same patching procedure, generating 256×256 patches, resulting in $1,000$ patches for the test set of the Lizard dataset.

In generating synthetic layouts, we aim to match the channel-wise cell counts observed in the real layouts of the test set. For each real test layout, we calculate the counts of each cell type across the channels and use these as conditional inputs during inference. This ensures that the generated synthetic patches exhibit similar cell count distributions to those observed in the real test layouts.

11. Evaluation Metrics

To evaluate the quality of the generated cell layouts and pathology images, we employ a set of metrics focusing on different aspects, such as visual fidelity, topological similarity, and utility to downstream tasks.

First, the **Fréchet Inception Distance (FID)** [30] measures visual similarity by comparing the distributions of features extracted from a pre-trained Inception network between real and generated images. Lower FID scores indicate greater visual realism in the generated images. Feature extraction is tailored to each dataset with custom-trained models. Here, the FID we used is the **spatial-FID** proposed in Spatial Diffusion [44]. The spatial-FID replaces visual features with a spatial representation derived from an autoencoder's intermediate layer, and we trained the autoencoder in the same way. In addition, we extended it to the Lizard dataset by training another autoencoder in the same manner. We also evaluate the accuracy of the generation through cell count error, calculating discrepancies between real and generated cell counts per type and overall. In our experiments, we use the connected component labeling method [58] to count the cell numbers. Assume there are n types of cells. For each cell type i, the **cell count error (CCE)** across N test samples is defined as:

$$Cell\_Count\_Error^{(i)} = \frac{1}{N}\sum_{j=1}^{N}\left|c_{real,j}^{(i)} - c_{syn,j}^{(i)}\right|$$

with **total count error (TCE)** calculated as:

$$Total\_Count\_Error = \frac{1}{N}\sum_{j=1}^{N}\left|\sum_{i=1}^{n} c_{real,j}^{(i)} - \sum_{i=1}^{n} c_{syn,j}^{(i)}\right|$$

where c is the cell count. In addition, our proposed **TopoFD** metric is used to evaluate the topological similarity between real and generated cell layouts. Lower TopoFD scores indicate closer alignment in spatial structure.

We also use the metric proposed in [66], Maximum Mean Discrepancy (MMD) [24] to measure the topological difference between the real and synthetic distributions. The persistence diagrams from synthetic and real layouts are embedded into a reproducing kernel Hilbert space (RKHS).

The MMD computes the distance between the mean embeddings of these two distributions in the RKHS. Given two sets of persistence diagrams, $\mathcal{D}_{syn} = \{Dgm_i^{syn}\}_{i=1}^N$ from the synthetic data and $\mathcal{D}_{real} = \{Dgm_j^{real}\}_{j=1}^N$ from the real data, we can define the mean of each diagram set,

$$\Phi(\mathcal{D}_{syn}) := \frac{1}{N}\sum_{i=1}^{N}\Phi(Dgm_i^{syn})$$

$$\Phi(\mathcal{D}_{real}) := \frac{1}{N}\sum_{j=1}^{N}\Phi(Dgm_i^{real})$$

Then, the MMD is defined as:

$$\text{MMD}(\mathcal{D}_{\text{syn}}, \mathcal{D}_{\text{real}}) := \|\Phi(\mathcal{D}_{\text{syn}}) - \Phi(\mathcal{D}_{\text{real}})\|_{\mathcal{H}}$$

In terms of the kernel for persistence diagrams, we use the Gaussian kernel based on the 1-Wasserstein distance between diagrams,

$$k_{W_1}(Dgm_i, Dgm_j) = \exp\left(-\frac{W_1(Dgm_i, Dgm_j)}{\sigma^2}\right)$$

Lastly, to enhance downstream utility, we used $2,000$ generated image-layout pairs as augmented training data for cell detection and classification tasks, evaluating their performance with the **F1-score**.

12. Cell Count Distribution Analysis

Also, to ensure the accurate generation of cell distributions, the training set encompasses a wide range of cell count values. As shown in Fig. 7, we randomly select $2,000$ patches during the training. We analyzed and observed each cell type's range of cell counts in the training patches to confirm coverage across typical values observed in test conditions. This observation is crucial for the diffusion model, as it needs exposure to the range of cell counts during training to accurately generate corresponding counts during the inference.

13. Additional Ablation Study

Ablation Study on learning rate. This ablation study examines the effect of different learning rates on model performance. The results indicate that a learning rate of 2×10^{-5} achieves the best overall performance across all metrics, with the lowest FID, Total Counting Error, and TopoFD values. Higher learning rates, such as 1×10^{-4}, result in a higher total counting error and TopoFD, suggesting that an overly large learning rate may hinder convergence. Conversely, lower learning rates, including 1×10^{-5} and 5×10^{-5}, show some improvements but do not reach the optimal balance across all metrics. The chosen learning rate of 2×10^{-5}, therefore, appears to provide the best trade-off, facilitating convergence that enhances both cell counting accuracy and fidelity in the synthetic cell layouts.

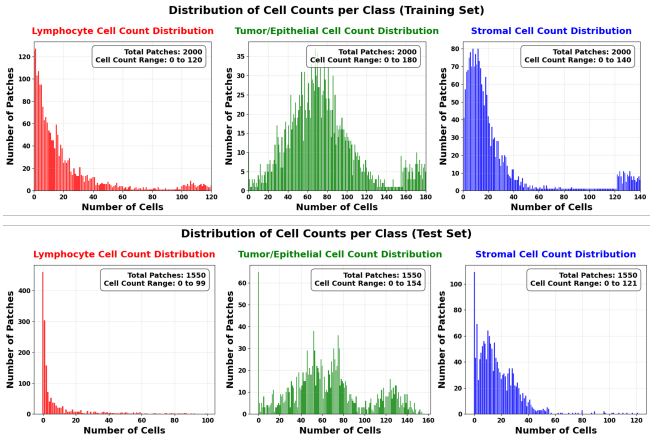

Figure 7. The statistical analysis of the cell count distributions on the BRCA-M2C training and test sets.

learning rate	BRCA-M2C		
	FID ↓	TCE ↓	TopoFD ↓
1e-4	0.021	12.357	75.667
1e-5	0.015	6.314	81.397
5e-5	0.066	12.367	85.949
2e-5	**0.005**	**5.192**	**69.354**

Table 5. Ablation study on learning rate.

14. Biological Plausibility

Specifically, we randomly selected 10 pairs of real and synthetic cell layouts as shown in Fig. 8. Without revealing their type (synthetic/real), we asked the expert to (1) identify which layout is synthetic; (2) characterize the tissue biology of these layouts. The expert achieves only a 60% accuracy in identifying the synthetic layout, confirming the realism of our synthetic layouts even to a domain expert. Regarding the characterization of tissue biology, as shown in Fig. 8, the pathologist concluded that for each pair of layouts, the synthetic layout preserved the defining biological characteristics of its corresponding real sample, consistently reflecting benign/low-grade or cancerous/high-grade properties. These experiments with a domain expert offer direct evidence, beyond quantitative measures and downstream analyses, that our generated layouts align well with actual biological structures.

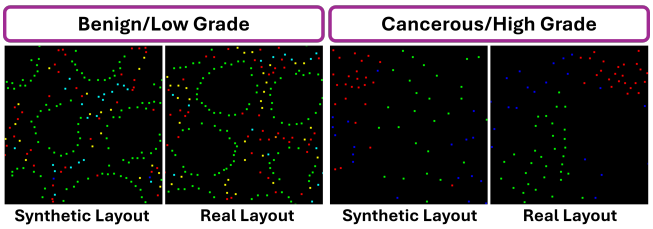

Figure 8. Biological plausibility validated by the domain expert.

15. Spatial Point Pattern Analysis

We also evaluate our synthetic layouts using one standard statistical method for spatial point pattern analysis. Specifically, we employ multivariate Ripley's K-functions to evaluate the synthetic layouts of the BRCA-M2C dataset, which comprises 3 cell types. For each test reference layout, we have a corresponding synthetic layout and extract cell centroids from both. We then compute 3 K-functions to capture intra-class clustering (one per cell type) and 6 cross-K functions to describe inter-class interactions. Next, we examine the difference between real and synthetic K-values over 6 radii: $[15, 30, 45, 60, 75, 90]$. For each radius and each cell-type pair, we perform a paired t-test to check if synthetic data deviates significantly from real layouts. This procedure yields 54 p-values (18 from intra-class and 36 from inter-class analyses). We then count the number of cases where these p-values exceed 0.05, indicating no statistically significant difference. Overall, as shown in Tab. 6 and Tab. 7, ***TopoCellGen*** achieves a greater number of radii with no significant difference is observed, compared to other methods. It most accurately produces both intra-class clustering and inter-class interactions, demonstrating the closest alignment with real data across the evaluated radii.

Method	BRCA-M2C			
	Lym. – Lym.	Epi. – Epi.	Stro. – Stro.	Total
ADM	0/6	0/6	2/6	2/18
TMCCG	2/6	1/6	2/6	5/18
Spatial Diffusion	1/6	3/6	2/6	6/18
TopoCellGen	3/6	5/6	4/6	12/18

Table 6. Number of radii with no statistically significant difference ($p > 0.05$) for intra-class spatial clustering.

Method	BRCA-M2C						
	Lym. – Epi.	Lym. – Stro.	Epi. – Lym.	Epi. – Stro.	Stro. – Lym.	Stro. – Epi.	Total
ADM	1/6	3/6	2/6	1/6	1/6	2/6	10/36
TMCCG	3/6	2/6	3/6	4/6	3/6	2/6	17/36
Spatial Diffusion	3/6	4/6	2/6	3/6	1/6	2/6	15/36
TopoCellGen	4/6	3/6	4/6	5/6	3/6	5/6	24/36

Table 7. Number of radii with no statistically significant difference ($p > 0.05$) for inter-class spatial interactions.

16. Computational Costs and Scalability

Currently, our model is trained on a single NVIDIA A6000 GPU with 48 GB of memory for approximately 360K steps, using a batch size of 5 at 256×256 resolution within 200 hours. The experiments can also be seamlessly scaled with data parallel training.

17. Limitations

Our proposed *TopoCellGen* will fail in some cases. First, the model is limited by its dependence on the cell types present in the training data, preventing it from generating layouts containing unseen cell types. Additionally, the model currently generates cell layouts in 256×256 patches, which constrains its application to small-scale regions.