DiffBulk: Enhancing Spatial Transcriptomic Prediction with Diffusion-Based Training

Bochong Zhang, Tianyi Zhang, Qiaochu Xue, Zeyu Liu, Dankai Liao, Timothy Antoni, YEO HUI TING GRACE, Sicheng Chen, Hwee Kuan LEE, Shangqing Lyu, and Yueming Jin, *Member, IEEE*

***Abstract*— Spatial Transcriptomics (ST) technology detects gene expression from tissue biopsies, playing an emerging role in cancer diagnosis and precision medicine. However, the high cost of ST technology limits its broader application. Recently, deep learning approaches have provided insight into predicting gene expression based on H&E-stained histopathology images. Nevertheless, the relationship between morphological features and gene expression is highly complex. To address these challenges, we propose DiffBulk, a novel two-stage framework that leverages conditional diffusion models to learn expressive image representations enriched with gene expression information. In the first stage, we introduce a gene-to-image conditional diffusion model equipped with a permutation-invariant open-embedding gene encoder, which enables unified training across diverse gene panels. In the second stage, diffusion-derived features are fused with representations from a pathology foundation model, effectively bridging the domain gap and improving downstream gene expression prediction. We evaluate DiffBulk on high-quality Xenium ST data curated from the HEST dataset and the CrunchDAO challenge, constructing tile-level pseudo-bulk datasets for training and evaluation. Extensive experiments demonstrate that DiffBulk consistently outperforms state-of-the-art baselines across all metrics for gene expression prediction. These findings highlight the potential of diffusion-based gene-image representation learning and suggest promising directions for future research.**

***Index Terms*— Gene Expression Prediction, Conditional Diffusion Model, Open-Embedding, Foundation Model.**

This work was supported by the Ministry of Education Tier 1 grant, Singapore (24-1250-P0001), and the Ministry of Education Tier 2 grant, Singapore (T2EP20224-0028). This work was powered by the UnPuzzle & PuzzleCloud Platform (https://puzzlelogic.com/unpuzzle) and supported by PuzzleLogic Pte Ltd, Singapore, and supported by the Agency for Science Technology and Research, Singapore under the Artificial Intelligence in Drug Discovery (AIDD) Programme (H25A1N0002).

Bochong Zhang and Tianyi Zhang contributed equally to this work. Corresponding Author: Shangqing Lyu (e-mail: shangqinglyu@puzzlelogic.com) and Yueming Jin (e-mail: ymjin@nus.edu.sg)

Bochong Zhang, Tianyi Zhang, Qiaochu Xue, and Yueming Jin are with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117417 (e-mails: {bochong, zhangtianyi, e1352520}@u.nus.edu; ymjin@nus.edu.sg).

Zeyu Liu, Dankai Liao, Sicheng Chen, and Shangqing Lyu are with PuzzleLogic Pte Ltd, Singapore 229594 (e-mails: {zeyuliu, dankailiao, sichencheng, shangqinglyu}@puzzlelogic.com).

Timothy Antoni and Hui Ting Grace Yeo are with the Genome Institute of Singapore (GIS), Agency for Science, Technology and Research (A*STAR), 60 Biopolis Street, Singapore 138672, Singapore (e-mails: timothy.antoni@u.nus.edu; grace_yeo@a-star.edu.sg).

Tianyi Zhang is also with Bioinformatics Institute (BII), Agency for Science, Technology and Research (A*STAR), Singapore 138671, Singapore (e-mail: zhangty@a-star.edu.sg)

Hwee Kuan Lee is with the Bioinformatics Institute (BII), Agency for Science, Technology and Research (A*STAR), Singapore 138671, Singapore, and the School of Computing, National University of Singapore, Singapore 117417, Singapore, School of Biological Sciences, Nanyang Technological University, Singapore 639798, Singapore, and International Research Laboratory on Artifical Intelligence, Singapore 138632, Singapore, and A* Centre for Frontier AI Research (CFAR), 1 Fusionopolis Way #16-16 Connexis (North Tower) Singapore 138632 (e-mail: leehk@a-star.edu.sg)

Yueming Jin is also with the Department of Biomedical Engineering, National University of Singapore, Singapore 117417, Singapore (e-mail: ymjin@nus.edu.sg)

I. Introduction

PATHOLOGICAL image analysis remains the gold standard in clinical practice for cancer diagnosis. By examining H&E-stained histology slides, pathologists assess tissue conditions based on morphological changes [1]. Beyond visual observation, profiling gene expression in tissues offers complementary molecular information, providing deeper insight into tumor biology [2]. While traditional transcriptomic techniques such as bulk [3], [4] and single-cell RNA sequencing [5], [6] offer comprehensive gene expression profiling, they inherently disrupt spatial organization due to tissue dissociation [7]. In contrast, spatial transcriptomics (ST) addresses this limitation by capturing gene expression at defined tissue locations, thus integrating molecular signals with histological context [8], [9]. To reduce technical noise, pseudo-bulk expression further aggregates gene signals from spatially adjacent regions [10], [11]. Unfortunately, ST assays remain expensive and labour-intensive, prompting intense interest in predicting gene expression directly from pathological images.

Recent advances in deep learning offer a trending possibility. Early methods such as ST-Net [12], DeepSpaCE [13], HisToGene [14], and Hist2ST [15] cast the image-to-ST task as a regression problem, directly mapping visual features to gene-expression values. To mitigate data scarcity, several studies pretrain foundation models (FMs) on large histopathology corpora via self-supervised learning and subsequently fine-tune them on limited gene-expression datasets [16]–[19]. However, the histology-to-transcriptomics mapping is inherently ill-posed: diverse morphological patterns can yield similar bulk expression profiles, whereas subtle visual differences may accompany pronounced transcriptional shifts. Accordingly, direct-regression models produce predictions that average across multiple plausible molecular states, thereby obscuring biologically meaningful spatial–gene co-patterns.

To address these challenges, two-stage frameworks have emerged as a promising solution. These methods decouple the learning process into a pretraining stage and a downstream prediction or regression stage. During pretraining, by explicitly incorporating gene modality, the model learns to capture the underlying relationship between gene expression profiles and histological features. This multimodal feature space reduces the ambiguity inherent in direct regression and guides the downstream predictor toward more plausible predictions. For example, BLEEP [20] trains both an image encoder and an MLP-based gene encoder using a contrastive learning objective. Then, they utilize the pretrained image encoder to retrieve the top-k most similar histological images from the reference dataset. Their gene expression values are averaged to produce the final prediction. However, contrastive learning assumes a clear semantic boundary between positive and negative pairs. In the context of gene expression prediction, this assumption fails, as visually similar histological images can exhibit different gene expression profiles. Treating near-positives as negatives introduces label noise, which misguides representation learning. Moreover, its MLP-based gene encoder is limited to fixed gene sets and fails to preserve the permutation invariance property-a fundamental property of gene expression data whereby swapping two genes (gene_i and gene_j) along with their expression values (v_i and v_j) should yield identical encodings. However, because the MLP's positional weights are fixed, permuting the input order of genes produces different encodings, thereby breaking permutation invariance.

Recently, diffusion-based pretraining has gained traction as a means to learn richer image representations. Prior work has shown its benefits across a variety of downstream tasks, including classification [21]–[24], segmentation [25]–[28], and semantic correspondence [29]–[31]. However, gene-to-image diffusion pretraining remains underexplored. Although it holds potential for bridging the modality gap between histological images and transcriptomic profiles, integrating heterogeneous gene sets into a single diffusion framework poses a key challenge: one must design a gene encoder that can flexibly handle arbitrary panels while preserving permutation invariance.

To address this gap, we introduce **DiffBulk**, a novel two-stage framework for tile-level pseudo-bulk gene expression prediction. In the first stage, a gene-to-image conditional diffusion model [32], [33] is trained on paired image–gene expression data. We introduce an open-embedding gene encoder that preserves permutation-invariance property across heterogeneous gene sets. To ensure that the model could complete the gene prediction task without gene condition, we introduce a Probabilistic Masking Switch (PMS) module, which randomly omits gene expression values with probability p. In the second stage, the pretrained conditional diffusion U-Net serves as a frozen backbone for gene prediction. Notably, we inject low-level noise into the histology tiles and pass them through the frozen diffusion model to obtain multi-scale intermediate activations. These activations are then fused by a Multi-Scale Feature Extraction Network (MSFE-Net) and integrated with fm-based features for final gene prediction.

Lastly, we evaluate DiffBulk on three high-quality Xenium ST datasets by creating tile-level pseudo-bulk from HEST [34] and CrunchDAO challenge [35]. Specifically, we construct the HEST-Bowel, HEST-Pancreas, and CrunchDAO-Bowel datasets, comprising 16,816, 7,501, and 13,021 pseudo-bulk image–gene expression pairs, respectively. We benchmark our method against four task-specific gene prediction models [12]–[14], [20] and three foundation models [16]–[18]. DiffBulk achieves state-of-the-art performance on all three datasets. Our main contributions are summarized as follows:

1) We propose **DiffBulk**, a novel two-stage framework for tile-level pseudo-bulk gene expression prediction. This paradigm naturally handles the one-to-many ambiguity and learns robust, diffusion-based image representations that bridge histology and transcriptomics.
2) To provide a flexible condition across arbitrary gene sets, we design an open-embedding gene encoder that preserves the permutation invariance of gene expression data. This design enhances the scalability of our framework to accommodate diverse and expanding gene–image datasets.
3) We introduce a Probabilistic Masking Switch (PMS) module during diffusion pretraining. PMS guarantees that the pretrained U-Net remains effective for downstream gene prediction.
4) We perform extensive evaluations on three tile-level pseudo-bulk ST datasets from HEST and CrunchDAO challenges. DiffBulk consistently outperforms existing task-specific and fm-based methods. The code is publicly available at `https://github.com/iMVR-PL/DiffBulk`.

II. Related Work

A. Image to Gene Expression Prediction

The task of predicting gene expression from histopathological images has received increasing attention in recent years. Early models such as ST-Net [12] and DeepSpaCE [13] adopt a transfer learning approach, where 224×224-pixel image patches centered on ST spots are extracted and passed through pretrained convolutional neural networks (DenseNet-121 [36] and VGG16 [37], respectively) followed by a linear regressor to predict gene expression values. To enhance predictive accuracy and better capture spatial dependencies, more recent models have introduced architectural innovations. HisToGene [14] combines Vision Transformers (ViTs) [38] with dynamic convolutional networks to capture long-range dependencies and local contextual cues within tissue sections. Similarly, Hist2ST [15] employs a convolutional mixer module to integrate local image textures with spatial gene expression patterns. While these approaches have demonstrated promising results, they are based on direct regression frameworks that overlook the ill-posed nature of the task, where histological features alone may be insufficient to fully recover gene expression profiles. To mitigate the ill-posed nature of direct regression, BLEEP [20] proposes a two-stage contrastive learning framework. It aligns image and gene embeddings during pretraining, then retrieves top-k similar reference tiles for prediction via expression averaging. However, contrastive

learning assumes clear semantic boundaries between positive and negative pairs, which rarely hold in this setting. Visually similar tiles may have divergent gene profiles, leading to noisy supervision. Furthermore, BLEEP's MLP-based gene encoder is limited to fixed gene sets and violates permutation invariance property of gene expression data. A closely related work is Stem [39], which employs a conditional diffusion model to directly model the distribution of gene expression conditioned on corresponding images. After training, gene expression predictions are obtained via the standard denoising process. While both Stem and our approach utilize conditional diffusion models, our work focuses on a different objective: rather than using diffusion for generation, we aim to explore and extract diffusion-based representations that encode rich image-gene relationships, thereby enhancing the prediction of pseudo-bulk gene expression.

B. Pathological Foundation Models

Inspired by recent breakthroughs in computer vision, large-scale pathology foundation models (FMs) have been developed to learn general-purpose visual representations. UNI [16] and GigaPath [17] pretrained large Vision Transformers on over 100 million and 1.3 billion pathology images, respectively, using a self-supervised strategy inspired by DINOv2 [40]. These models demonstrate remarkable transferability across various downstream tasks such as tumor classification and subtyping. Other efforts have explored multi-modal alignment strategies. PLIP [18] employs contrastive learning on 208,414 paired pathology image–text samples, while CONCH [19] adapts the CoCa framework [19] to jointly model visual and linguistic representations, enabling flexible cross-modal reasoning. Despite their strong image understanding capabilities, these foundation models are trained without incorporating genomic information. Consequently, the learned visual representations may not align well with transcriptomic signals, resulting in a domain gap when applied to gene expression prediction. This task misalignment limits the direct applicability of general-purpose FMs in gene modeling tasks, especially under data-scarce conditions.

C. Diffusion-based Representations

Several recent studies have explored the internal representations and applications of diffusion models in biomedical imaging. Recent works such as SDSeg [41], GM-SDE [42], and MSDiff [43] extend diffusion modeling to segmentation and image reconstruction tasks, demonstrating the versatility of diffusion priors for learning structural and spatial representations efficiently. Beyond these applications, Diffusion Classifier [22] and DDAE [21] extract features from pretrained diffusion models for image classification, achieving performance comparable to contrastive learning [44] and masked autoencoders [45]. DDPMSeg [26] and ODISE [25] aggregate features from carefully selected layers and timesteps to support semantic and panoptic segmentation, respectively. For semantic correspondence, Zhang et al. [29] integrate diffusion-based features with DINOv2 representations to improve spatial alignment. Diffusion Hyperfeatures [30] further enhance this line of work by consolidating multi-scale and multi-timestep feature maps into dense descriptors. Although diffusion-based representations have shown strong performance across diverse tasks, their ability to model the complex and ill-posed relationship between histological images and gene expression remains underexplored, and effectively incorporating heterogeneous gene sets as conditions in the diffusion framework remains a challenge.

III. Methodology

A. Problem Formulation

To enable tile-level gene expression prediction, we curate high-quality Xenium data to construct corresponding pseudo-bulk expression profiles with the corresponding histopathology image tiles. In particular, H&E-stained WSIs at $20X$ magnification are divided into several non-overlapping image tiles $\mathbf{x} \in \mathbb{R}^{w \times h}$, where w and h denote the width and height (in pixels) of each tile. For each tile, we aggregate the corresponding pseudo-bulk gene expression $\mathbf{g} \in \mathbb{R}^{n_1}$, where n_1 denotes the number of gene types. To ensure data reliability, we apply expression normalization and quality control procedures, including variance-based filtering of gene expression and image feature filtering as described in [34]. This preprocessing yields a pseudo-bulk tile dataset composed of paired samples $(\mathbf{x}, \mathbf{g})$, suitable for training models to predict gene expression from visual pathological features.

Formally, the goal is to learn a function $f_\theta : \mathbb{R}^{w \times h} \to \mathbb{R}^{n_1}$, parameterized by θ, that maps each image tile $\mathbf{x}$ to its corresponding gene-expression vector $\mathbf{g}$. This problem is ill-posed, as visually similar tiles can have different gene expression profiles. To overcome the shortcomings of direct regression, we propose a diffusion-based two-stage framework. First, we model the conditional distribution $p(\mathbf{x}|\mathbf{g})$ to capture the full diversity of morphology given gene expression; then, we perform regression on individual genes. This strategy preserves complex spatial–gene co-patterns that direct regression would otherwise average out.

B. DiffBulk Framework

As illustrated in Figure 1, DiffBulk comprises two training stages: gene-to-image conditional diffusion training and downstream gene expression prediction. In the conditional diffusion training stage, we introduce an open-embedding gene encoder capable of preserving permutation-invariant property. This encoder is integrated with a gene-to-image diffusion model, which implicitly captures the complex relationship between gene expression and their corresponding histological image tiles. In the downstream stage, diffusion-derived features $\mathbf{z}_{\text{diff}}$ from the pretrained U-Net and foundation model features $\mathbf{z}_{\text{fm}}$ from an fm encoder are fused via a gated module to produce an enhanced representation $\mathbf{z}$, which is used to predict pseudo-bulk gene expression.

1) Open-Embedding Gene Encoder: Existing MLP-based gene encoders require a fixed input size and violate the permutation-invariant property of gene expression data. To address these limitations, we propose an open-embedding gene encoder $c_{\text{gene}}(\cdot)$, which supports arbitrary gene sets and

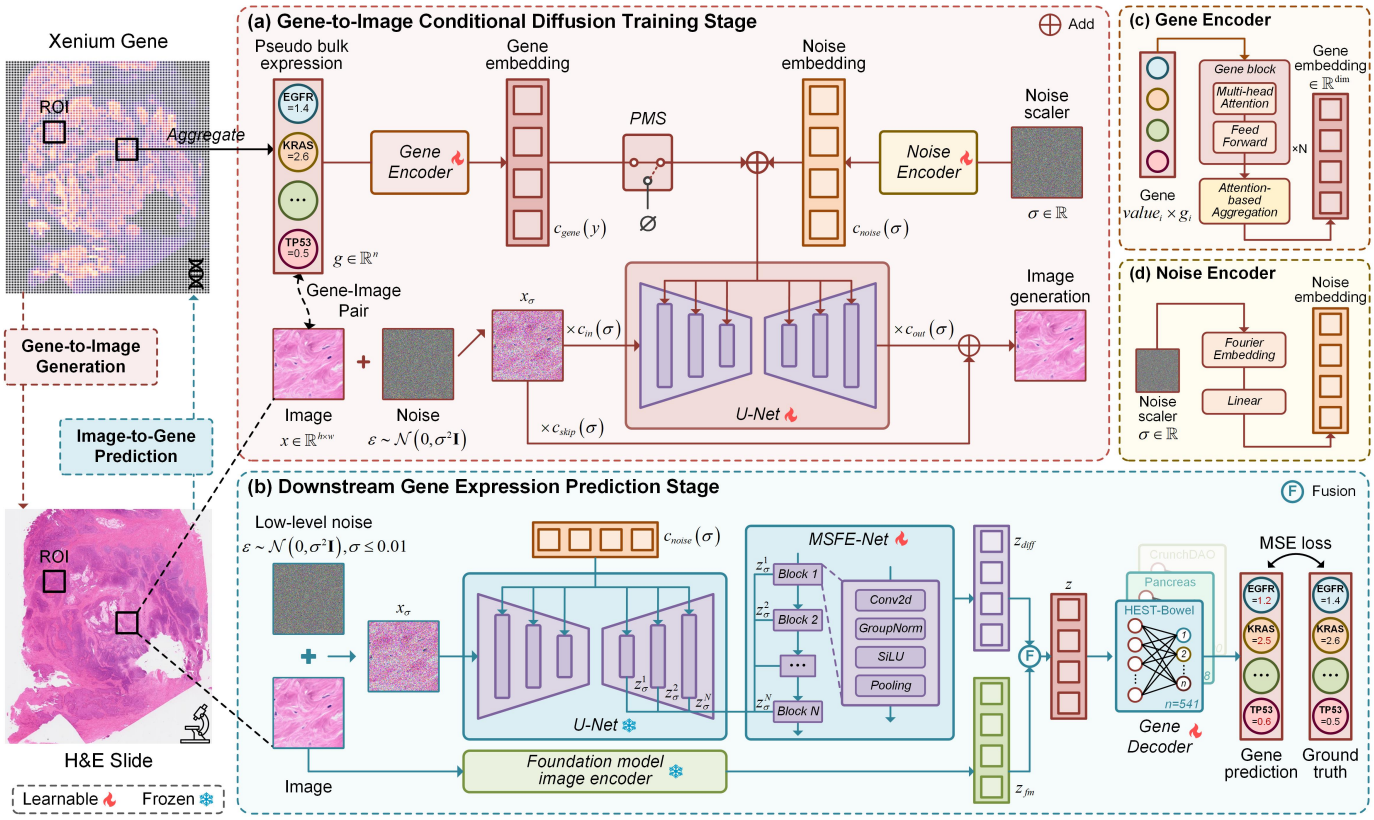

Fig. 1. Overview of the proposed **DiffBulk** framework. The part (a) illustrates the gene-to-image conditional diffusion training stage, where a permutation-invariant open-embedding gene encoder guides the generation of pathology image features. The part (b) shows the downstream pseudo-bulk gene expression prediction stage, where diffusion-based features are fused with FM features via a gated fusion module for gene expression prediction.

preserves structural invariance. As illustrated in Figure 1, each gene type is treated as a token with a learnable embedding $g_i \in \mathbb{R}^{\text{dim}}$. All embeddings are collected in a global gene vocabulary matrix $V \in \mathbb{R}^{N\times\text{dim}}$, where N is the total number of unique genes across the training cohorts. This design allows seamless incorporation of diverse gene panels from multiple sources. To integrate expression magnitude, we compute an expression-aware gene embedding emb_i by taking the element-wise product between the gene encoding g_i and its corresponding expression value $v_i \in \mathbb{R}$.

Given a gene expression profile from a specific tissue region or tile, consisting of n_1 gene-expression pairs $\{(g_1, v_1), \ldots, (g_{n_1}, v_{n_1})\}$, we construct the input embedding $\text{Emb}_{\text{in}} \in \mathbb{R}^{n_1\times\text{dim}}$ by stacking the expression-aware embeddings emb_i. To guarantee permutation invariance, we then pass Emb_{in} through a transformer-based gene encoder (no positional encodings), leveraging the fact that vanilla transformer [46] architectures without positional encoding are order-agnostic. After n identical blocks in the encoder, we obtain the contextualized output $\text{Emb}_{\text{out}} \in \mathbb{R}^{n_1\times\text{dim}}$. Finally, we introduce a learnable query vector $q \in \mathbb{R}^{1\times\text{dim}}$ and apply an attention-based aggregation over the n_1 token embeddings to produce a single unified gene representation. The aggregation process is formulated as

$$K = W_k \cdot \text{Emb}_{\text{out}}, V = W_v \cdot \text{Emb}_{\text{out}}, \tag{1}$$

$$y = \text{Softmax}(\frac{qK^T}{\sqrt{n_1}})V, \tag{2}$$

where $y \in \mathbb{R}^{1\times dim}$ represents the final aggregated gene embedding. With the gene encoder $c_{\text{gene}}(\cdot)$, we are able to perform joint training across multiple datasets with varying gene sets and explore a unified and expressive embedding space for gene features.

2) Two-Stage Training Framework: We propose a two-stage training framework to address the limitations of direct regression and contrastive pretraining approaches. In the first stage, to capture the complex relationship between histological images and corresponding gene expression profiles, we introduce a conditional diffusion training strategy. Our framework is built upon a modified score-based diffusion model with a U-Net backbone, following the EDM2 paradigm [47]. This formulation enables the model to learn rich pathology representations conditioned on gene expression vectors, effectively bridging the modality gap between transcriptomic and visual domains.

Our approach adopts a σ-dependent skip connection strategy, which allows the network to dynamically adapt to different levels of signal corruption during the denoising process. This mechanism enables the model to interpolate between two objectives: recovering the clean image $\mathbf{x}$ when the noise level σ is low, and estimating the injected noise component ϵ when σ is high. Formally, the conditional denoising function

is expressed as:

$$D_{\theta_1}(\mathbf{x}_\sigma;\sigma,\mathbf{g}) = c_{\text{skip}}(\sigma)\,\mathbf{x}_\sigma + c_{\text{out}}(\sigma)\,F_{\theta_1}\big(c_{\text{in}}(\sigma)\,\mathbf{x}_\sigma;\, c_{\text{noise}}(\sigma),\, c_{\text{gene}}(\mathbf{g})\big), \tag{3}$$

where $\mathbf{x}_\sigma = \mathbf{x} + \epsilon$ is the noisy image input, and $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$ is the Gaussian noise. F_{θ_1} denotes the U-Net-based denoising network. The modulation functions $c_{\text{in}}(\sigma)$, $c_{\text{skip}}(\sigma)$, and $c_{\text{out}}(\sigma)$ are scale-dependent coefficients that modulate the input signal, skip connection, and output contribution, respectively, based on the noise level σ. In addition, $c_{\text{noise}}(\sigma)$ and $c_{\text{gene}}(\mathbf{g})$ provide conditional embeddings for the noise level and gene expression profile, respectively.

The denoising task enables the model to leverage molecular information for generating visual features. The skip connection, scaled by $c_{\text{skip}}(\sigma)$, directly injects the noisy input $\mathbf{x}_\sigma$ into the output. This process contributes to preserving structural information, particularly when $sigma$ is small and much of the original image signal remains intact. In contrast, the residual term produced by the denoising network F_θ provides a noise-aware, gene-conditioned refinement that becomes increasingly dominant as the noise level increases.

To ensure that the model remains effective in the downstream gene prediction task, where gene conditions are unavailable, we introduce the Probabilistic Masking Switch (PMS) module. During diffusion pretraining, PMS randomly drops the gene condition with a fixed probability p, encouraging the model to learn both conditional and unconditional representations. The masking operation is defined as:

$$y = \begin{cases} \varnothing & \text{with probability } p, \\ c_{\text{gene}}(\mathbf{g}) & \text{otherwise.} \end{cases} \tag{4}$$

where $c_{\text{gene}}(\mathbf{g})$ denotes the gene embedding derived from the expression profile $\mathbf{g}$, and $\varnothing$ denotes the absence of any gene condition. This design facilitates effective gene expression prediction in the subsequent stage.

In the downstream stage, we aim to extract and refine diffusion-based features by integrating them with FM-based features. To this end, we introduce MSFE-Net (Multi-Scale Feature Extraction Network), a lightweight feature adapter designed to extract diffusion-based representations, denoted as $\mathbf{z}_{\text{diff}}$. MSFE-Net operates on the intermediate activations of the U-Net, which was trained during the pretraining stage and remains frozen in this phase.

To maintain alignment with the denoising paradigm during diffusion pretraining, we inject low-level Gaussian noise ϵ into the input pathology image, producing a perturbed image $\mathbf{x}_\sigma \in \mathbb{R}^{w\times h}$. This noised image is then fed into the frozen U-Net without gene conditioning. The network produces a set of multi-scale feature activations $\mathcal{Z}_\sigma = \{z_\sigma^1, z_\sigma^2, \ldots, z_\sigma^N\}$, where each z_σ^i is extracted from the i-th block of the U-Net. These activations encode varying levels of semantic and structural information, influenced by both the input noise level and the hierarchical depth of the network. In practice, we select a subset $\mathcal{S} = \{i_1, i_2, \ldots, i_s\} \subseteq \{1, \ldots, N\}$ of blocks, typically focusing on decoder blocks that are known to capture higher-level semantic features, and apply MSFE-Net to each selected activation.

$$f_\sigma^{i_j} = \text{MSFE-Net}(z_\sigma^{i_j}), \quad j = 1, \ldots, s. \tag{5}$$

The resulting multi-scale features are then concatenated to construct the final diffusion-based representation $\mathbf{z}_{\text{diff}} = \text{Concat}(f_\sigma^{i_1}, f_\sigma^{i_2}, \ldots, f_\sigma^{i_s})$. To enhance the expressiveness of $\mathbf{z}_{\text{diff}}$, we introduce a gated fusion mechanism to effectively integrate diffusion-derived features with FM-extracted visual features. Let $\mathbf{z}_{\text{fm}} = \text{FM}(\mathbf{x})$ denote the visual features extracted from the frozen FM encoder given an input image $\mathbf{x}$. The fusion process is defined as:

$$g = \sigma\left(\text{Linear}\left(\text{Concat}(\mathbf{z}_{\text{fm}}, \mathbf{z}_{\text{diff}})\right)\right), \tag{6}$$

$$\mathbf{z} = \mathbf{z}_{\text{fm}} + g \odot \mathbf{z}_{\text{diff}}, \tag{7}$$

where $\text{Linear}(\cdot)$ represents a fully connected layer, $\sigma(\cdot)$ is the sigmoid activation function that bounds the gate values within $[0, 1]$, and $\odot$ denotes element-wise multiplication. This gating mechanism enables the model to dynamically weigh the contribution of diffusion-derived features according to image content. The enhanced feature $\mathbf{z}$ is then passed through a linear projection layer to predict gene expression levels.

C. Objective Function

In the gene conditional diffusion training stage, the objective is formulated based on denoising score matching across varying noise levels. For a fixed noise level σ, the training loss is defined as:

$$\mathcal{L}(D_{\theta_1};\sigma) = \mathbb{E}_{(\mathbf{x},\mathbf{g})\sim p_{\text{data}},\epsilon\sim\mathcal{N}(0,\sigma^2 I)}\Big[\big\|D_{\theta_1}(\mathbf{x}_\sigma;\sigma,\mathbf{g}) - \mathbf{x}\big\|_2^2\Big], \tag{8}$$

where $\mathbf{x}$ and $\mathbf{g}$ denote paired pathology images and gene expression vectors, respectively, and ϵ represents Gaussian noise added to the input image. The perturbed input is given by $\mathbf{x}_\sigma = \mathbf{x} + \epsilon$. The total training objective is computed by integrating over a range of noise scales, weighted by a predefined function $\lambda(\sigma)$:

$$\underset{\theta_1}{\text{argmin}}\,\mathcal{L}(\theta_1) = \mathbb{E}_\sigma\big[\lambda(\sigma)\mathcal{L}(D_{\theta_1};\sigma)\big], \tag{9}$$

where $\lambda(\sigma)$ balances contributions from different noise levels and guides the model to perform well across the entire noise spectrum.

In the second training stage, after obtaining the enhanced image representation $\mathbf{z}$, the model predicts gene expression levels via a decoder. This decoder is implemented as a linear projection layer tailored to the number of target genes:

$$\hat{\mathbf{y}}_{\theta_2} = f_{\text{decoder}}(\mathbf{z}), \tag{10}$$

where θ_2 denotes the learnable parameters of the downstream network, including MSFE-Net, the decoder, and the gated fusion module. The training objective minimizes the mean squared error (MSE) between the predicted and ground truth gene expression:

$$\underset{\theta_2}{\text{argmin}}\,\mathcal{L}(\theta_2) = \frac{1}{n_1}\|\mathbf{y} - \hat{\mathbf{y}}_{\theta_2}\|_2^2 \tag{11}$$

IV. Experiments

A. Datasets

We evaluate the effectiveness of our proposed method using three publicly available tile-level pseudo-bulk datasets: two from the HEST project [34], covering distinct tissue types, and one from the CrunchDAO challenge [35]. We refer to them as *HEST-Bowel*, *HEST-Pancreas*, and *CrunchDAO-Bowel*, consisting of 3, 3, and 7 WSIs, respectively. The Xenium panels for the three datasets contain 541 (HEST-Bowel), 538 (HEST-Pancreas), and 460 (CrunchDAO-Bowel) genes, respectively. The overlap of gene types between any two datasets ranges from approximately 20% to 40%.

Each WSI was preprocessed by tiling H&E-stained histology slides at 20X magnification into non-overlapping patches of size 224×224 pixels. For each patch, gene expression profiles were aggregated from the corresponding Xenium spatial transcriptomics regions. Specifically, we first collected all spots spatially located within a patch, then aggregated their gene counts and applied a $\log(1+x)$ normalization to obtain the pseudo-bulk expression value associated with each tile. We further filtered out tiles with poor illumination or low-expression variance to ensure data quality. After preprocessing, we obtained 16,816, 7,501, and 13,021 paired image–gene samples for HEST-Bowel, HEST-Pancreas, and CrunchDAO-Bowel, respectively. We further explored three prediction settings based on expression variance thresholds, corresponding to the top-100, top-200, and all-gene subsets.

To avoid data leakage, all 3-fold cross-validation splits were performed strictly at the patient or slide level. For the HEST datasets, each fold uses two WSIs for training and the remaining one for testing. For the CrunchDAO-Bowel dataset, the seven WSIs are grouped into folds of 3, 2, and 2 slides, respectively. In all cases, tiles from the same WSI never appear across folds. All hyperparameters were kept fixed across folds to ensure consistency and reproducibility.

We report the Pearson Correlation Coefficient (PCC), Mean Absolute Error (MAE), and Mean Squared Error (MSE) as quantitative metrics. Results are expressed as $\mu \pm \sigma$, where μ and σ denote the mean and standard deviation computed over all test samples.

B. Training DiffBulk

1) Training Procedure: In the gene-to-image conditional diffusion training stage, we adopt EDM2 [47] as the backbone of our conditional diffusion model. The U-Net architecture consists of a symmetric encoder–decoder structure, each composed of four scale blocks operating at spatial resolutions of 224×224, 112×112, 56×56, and 28×28. To reduce computational cost, self-attention is applied only at the lowest resolution (28×28). Gene expression profiles are incorporated via our transformer-based open-embedding encoder, enabling unified training across HEST-Bowel, HEST-Pancreas, and CrunchDAO-Bowel datasets. It contains two identical transformer-encoder blocks. Furthermore, we set the hyperparameter $p = 0.5$ following the formulation of the PMS. The noise level σ is sampled from the log-normal distribution $\ln(\sigma) \sim \mathcal{N}(P_{\text{mean}}, P_{\text{std}})$, with $P_{\text{mean}} = -0.4$ and $P_{\text{std}} = 1.0$. The modulation functions were employed as follows: $c_{\text{skip}}(\sigma) = \frac{\sigma_{\text{data}}^2}{\sigma^2+\sigma_{\text{data}}^2}$, $c_{\text{out}}(\sigma) = \frac{\sigma \cdot \sigma_{\text{data}}}{\sqrt{\sigma^2+\sigma_{\text{data}}^2}}$, $c_{\text{in}}(\sigma) = \frac{1}{\sqrt{\sigma^2+\sigma_{\text{data}}^2}}$ and $c_{\text{noise}} = \frac{1}{4}\ln(\sigma)$, where $\sigma_{\text{data}} = 0.5$ denotes the expected standard deviation of the training data.

In the downstream stage, we focus on predicting pseudo-bulk gene expression for one of the HEST Bowel, HEST Pancreas, or CrunchDAO Bowel datasets. The U-Net model is kept frozen and serves as a feature extractor. We extract multi-scale decoder activations from the U-Net and feed them into the MSFE-Net to obtain the diffusion-based representation z_{diff}. In parallel, we employ the PLIP image encoder [18] to extract an fm-based feature z_{fm}. To maintain consistency with the denoising paradigm, we inject low-level Gaussian noise with standard deviation $\sigma = 0.01$ into the input image during both training and inference. The features z_{diff} and z_{fm} are then integrated via a gated fusion module, producing the final joint representation z. This representation is subsequently passed through a linear decoder to predict the gene expression levels.

2) Implementation Details: All experiments were implemented using PyTorch [48]. In the gene-to-image diffusion training stage, we trained on two NVIDIA GeForce RTX A6000 GPUs. The optimizer used was Adam [49], with $\beta_1 = 0.9$ and $\beta_2 = 0.99$, along with an inverse square root learning rate schedule. The initial learning rate was set to 0.01, and a post-hoc exponential moving average (EMA) with a decay length of 0.100 was applied to stabilize training performance. For the downstream training, we utilized a single NVIDIA GeForce RTX A6000 GPU. The optimization was conducted using the AdamW optimizer [50], with a learning rate of 0.0001 and a weight decay of 0.00001.

3) Efficiency and Computational Complexity: To assess the computational efficiency of DiffBulk, we analyze the training and inference cost in terms of model size, FLOPs, and runtime. The conditional diffusion U-Net with the integrated gene encoder in DiffBulk contains approximately 122.4M parameters, resulting in approximately 25.66 GFLOPs per 224×224 tile. In comparison, the BLEEP baseline includes 24.3M parameters and 4.11 GFLOPs, while the Gigapath foundation model contains 1.13B parameters and 223.45 GFLOPs per tile. Although diffusion pretraining introduces a moderate increase in computational cost compared with lightweight baselines, it remains substantially more efficient than fine-tuning large-scale foundation models. The conditional diffusion pretraining stage takes approximately 20 hours on two NVIDIA RTX A6000 GPUs, whereas fully fine-tuning Gigapath typically requires over 30 hours.

In the downstream stage, the diffusion backbone is frozen, and the lightweight MSFE-Net contains only 2.5M trainable parameters, making the training process highly efficient. During inference, DiffBulk combines diffusion-derived representations with a foundation-model branch, which naturally results in additional computational overhead. Specifically, DiffBulk processes 100 tiles in approximately 4000 ms, corresponding to about 7 minutes per WSI containing 10,000 tiles. In contrast, Gigapath alone processes 100 tiles in 3700 ms, or roughly 6 minutes per WSI. Thus, DiffBulk increases inference time by only 16.7%, a relatively small cost in real-world

pathology workflows, where slide-level turnaround times are dominated by upstream imaging and scanning stages rather than per-tile computation. Overall, these results demonstrate that DiffBulk provides meaningful gains with only modest additional computational cost, achieving a favorable balance between efficiency and accuracy.

C. Comparison with State-of-the-arts

1) Quantitative Results: We compared the performance of the proposed DiffBulk framework with several state-of-the-art approaches, including six task-specific models (ST-Net [12], DeepSpaCE [13], HisToGene [14], BLEEP [20]), LOKI [51] and STPath [52], three pathology foundation models (PLIP [18], UNI [16], and Gigapath [17]), as well as their fine-tuned variants. To assess the statistical significance, we further conducted paired one-sided t-tests under the same 3-fold slide-level split, comparing DiffBulk against each baseline. The quantitative results on the HEST-Bowel, HEST-Pancreas, and CrunchDAO-Bowel datasets are summarized in Table I.

As shown in Table I, DiffBulk consistently outperformed both task-specific and FM-based approaches across all three datasets. In the all-gene prediction setting, DiffBulk improved PCC by approximately 3%–5% on the HEST-Bowel and HEST-Pancreas datasets compared to FMs, and by 4%–6% compared to task-specific models. On the CrunchDAO-Bowel dataset, the gains were relatively modest, with PCC improvements of 1% over FMs and 3%–5% over task-specific methods. The corresponding statistical test results indicate that the performance improvements on HEST-Bowel and HEST-Pancreas are statistically significant, whereas the differences observed on CrunchDAO-Bowel are generally not significant. To further understand this smaller performance margin, we analyzed the statistical characteristics of the target gene expression profiles across datasets. Specifically, we computed the per-gene variance across all tiles within each dataset and observed that the CrunchDAO-Bowel dataset displays markedly lower expression variability. This indicates that a large proportion of genes in this dataset exhibit minimal variation across tiles, rendering the prediction task inherently easier and substantially less informative. Under such low-variance conditions, even simple baselines can approach optimal performance by approximating dataset-level mean expression values, thus leaving limited room for further improvement. Despite this ceiling effect, DiffBulk still achieves consistently lower MAE and MSE and higher PCC, demonstrating strong robustness and generalization.

Overall, DiffBulk effectively bridges the gap between task-specific models and large-scale foundation models. While FMs provide strong generic visual priors, they lack gene-aware inductive bias. Conversely, Task-specific models capture domain knowledge but struggle with cross-dataset scalability. DiffBulk overcomes these limitations by introducing a gene-to-image conditional diffusion pretraining strategy and a permutation-invariant open-embedding gene encoder. This design not only preserves the structural properties of gene expression data but also enables flexible integration of diverse gene sets, alleviating the scalability bottlenecks of existing models. As a result, DiffBulk generalizes more effectively across diverse datasets and consistently outperforms prior methods in gene expression prediction.

2) Qualitive Results: We visualize the prediction results for the CD24 gene in Figure 2, as CD24 plays a critical role in intestinal biology. Its expression has been linked to tumorigenesis and immune responses in colorectal cancer [53], making it a gene of considerable interest. Compared to other methods, DiffBulk more accurately identifies regions of high CD24 expression. In Figure 3, we further cluster tissue spots on an unseen bowel WSI based on their predicted gene expression profiles. The left panel includes partial expert annotations delineating key anatomical structures. Notably, DiffBulk effectively distinguishes regions such as the lamina propria (LP, blue) and muscularis propria (MP, orange), outperforming baselines like BLEEP and PLIP, and even compensating for gaps in the human-provided annotations. By delineating these structures solely from predicted transcriptomic profiles, DiffBulk demonstrates strong potential for automated tissue segmentation, offering a path toward reduced manual workload in histopathological analysis.

D. Effectiveness of Key Components

To thoroughly evaluate the effectiveness of each component in the DiffBulk framework, we perform a series of ablation studies. Unless otherwise stated, all ablated variants are trained on a single fold of the HEST-Bowel dataset and evaluated on the corresponding held-out test set. We compare four configurations: (1) training a U-Net from scratch and combining it with a PLIP [18] branch as a baseline; (2) replacing the U-Net with our pretrained gene-to-image diffusion model and using the proposed open-embedding gene encoder; (3) adding the Probabilistic Masking Switch (PMS) with masking probability $p = 0.5$ during diffusion pretraining; (4) injecting low-level Gaussian noise ($\sigma = 0.01$) during the downstream stage.

As shown in Table II, the U-Net trained from scratch achieved 1.070 MAE, 1.905 MSE, and 0.411 PCC. When initializing the U-Net with weights obtained from the gene-to-image diffusion training stage, the model achieved improved performance across all three metrics, with MAE and MSE reduced by 17% to 30% and PCC increased by 2.4%, reaching 0.898 MAE, 1.603 MSE, and 0.435 PCC. Incorporating the PMS module led to further improvements, resulting in 0.875 MAE, 1.583 MSE, and 0.442 PCC. Furthermore, injecting low-level noise during the downstream training phase yielded the best performance, with 0.864 MAE, 1.551 MSE, and 0.458 PCC.

E. Effectiveness of the Open-Embedding Gene Encoder

To assess the gene encoder in DiffBulk, we design a 3-layer MLP-based counterpart. Specifically, in the gene-to-image diffusion training stage, both encoders are trained under identical settings for fair comparison. As shown in Table III, the open-embedding gene encoder consistently achieves better performance than the MLP-based counterpart, owing to its

TABLE I

QUANTITATIVE COMPARISON WITH OTHER METHODS (MEAN ± STD OVER THE SAME 3-FOLD SLIDE-LEVEL SPLIT). † INDICATES THE FINE-TUNED VARIANT OF EACH FOUNDATION MODEL. THE BEST PERFORMANCE FOR EACH SETTING IS HIGHLIGHTED IN **BOLD**. STATISTICAL SIGNIFICANCE OF OURS OVER EACH BASELINE IS ANNOTATED BY PAIRED ONE-SIDED t-TESTS (H_0: NO DIFFERENCE; H_1: OURS IS BETTER) USING FOLD-LEVEL SCORES: $^{*}p < 0.1$, $^{**}p < 0.05$.

Dataset	Image Encoder	TOP-100			TOP-200			ALL		
		MAE ↓	MSE ↓	PCC ↑	MAE ↓	MSE ↓	PCC ↑	MAE ↓	MSE ↓	PCC ↑
HEST-Bowel	ST-Net [12]	1.935** ± 0.447	5.311** ± 2.401	0.328** ± 0.019	1.373** ± 0.244	2.971** ± 1.072	0.464** ± 0.075	1.114* ± 0.146	2.418** ± 0.599	0.400** ± 0.064
	DeepSpaCE [13]	1.536** ± 0.215	3.704** ± 0.934	0.391** ± 0.037	1.310* ± 0.103	2.733* ± 0.394	0.460** ± 0.035	1.128* ± 0.113	2.482** ± 0.435	0.400** ± 0.038
	HisToGene [14]	2.004** ± 0.259	5.270** ± 0.479	0.311** ± 0.032	1.381* ± 0.152	3.043* ± 1.140	0.431** ± 0.023	1.243** ± 0.141	2.272* ± 0.521	0.382** ± 0.052
	BLEEP [20]	1.548* ± 0.272	4.047** ± 1.128	0.366** ± 0.130	1.589** ± 0.477	4.128** ± 2.247	0.366** ± 0.143	1.162* ± 0.088	2.579* ± 0.317	0.396** ± 0.015
	Loki [51]	1.523* ± 0.235	3.810* ± 1.312	0.404** ± 0.042	1.406** ± 0.181	8.608** ± 1.084	0.150** ± 0.099	1.070 ± 0.188	2.324 ± 0.413	0.402** ± 0.088
	STPath [52]	1.493* ± 0.129	3.406* ± 0.744	0.417* ± 0.056	1.298 ± 0.174	2.705 ± 0.734	0.495 ± 0.046	1.074 ± 0.044	2.225 ± 0.574	0.422* ± 0.057
	PLIP [18]	1.492* ± 0.124	3.465* ± 0.562	0.408** ± 0.015	**1.283 ± 0.073**	**2.596 ± 0.331**	0.465* ± 0.040	1.060 ± 0.183	2.215* ± 0.596	0.412* ± 0.017
	PLIP† [18]	1.793** ± 0.209	4.406** ± 1.164	0.370** ± 0.071	1.308 ± 0.403	2.747 ± 1.374	0.494 ± 0.251	1.097 ± 0.099	2.271 ± 0.511	0.382** ± 0.034
	UNI [16]	1.646** ± 0.077	3.945** ± 0.296	0.334** ± 0.063	1.385* ± 0.083	2.905* ± 0.268	0.416** ± 0.076	1.091* ± 0.151	2.285* ± 0.582	0.403** ± 0.003
	UNI† [16]	1.609** ± 0.318	3.945** ± 1.427	0.336** ± 0.073	1.547** ± 0.412	3.728** ± 1.958	0.400** ± 0.086	1.158* ± 0.040	2.549* ± 0.507	0.427* ± 0.074
	Gigapath [17]	1.696** ± 0.179	4.072** ± 0.702	0.358** ± 0.024	1.432* ± 0.162	3.031* ± 0.562	0.430** ± 0.062	1.080* ± 0.139	2.218* ± 0.544	0.420* ± 0.013
	Gigapath† [17]	1.591** ± 0.191	3.981** ± 0.826	0.403** ± 0.038	1.334* ± 0.174	2.753 ± 0.670	0.481* ± 0.051	1.120* ± 0.090	2.367** ± 0.384	0.424* ± 0.031
	Ours	**1.478 ± 0.259**	**3.366 ± 1.105**	**0.462 ± 0.067**	1.308 ± 0.243	2.698 ± 0.934	**0.501 ± 0.071**	**1.049 ± 0.156**	**2.182 ± 0.507**	**0.435 ± 0.002**
HEST-Pancreas	ST-Net [12]	1.635** ± 0.147	4.043** ± 0.298	0.226** ± 0.126	1.357** ± 0.125	2.894** ± 0.294	0.331** ± 0.088	0.795** ± 0.107	1.359** ± 0.205	0.641** ± 0.059
	DeepSpaCE [13]	1.568** ± 0.201	3.659** ± 0.572	0.245** ± 0.097	1.324** ± 0.162	2.704** ± 0.445	0.330** ± 0.082	0.780** ± 0.132	1.313** ± 0.280	0.652** ± 0.060
	HisToGene [14]	1.644** ± 0.321	4.019** ± 0.255	0.233** ± 0.120	1.359** ± 0.138	2.764** ± 0.323	0.332** ± 0.062	0.803** ± 0.153	1.329** ± 0.216	0.643** ± 0.049
	BLEEP [20]	1.633** ± 0.098	4.143** ± 0.281	0.183** ± 0.080	1.366** ± 0.107	3.012** ± 0.321	0.317** ± 0.055	0.805** ± 0.087	1.396** ± 0.148	0.651** ± 0.063
	Loki [51]	1.613** ± 0.699	3.810** ± 4.585	0.224** ± 0.065	1.360** ± 0.367	3.033** ± 1.034	0.313** ± 0.063	0.854** ± 0.346	1.516** ± 1.515	0.620** ± 0.060
	STPath [52]	1.553** ± 0.268	3.403** ± 0.946	0.282** ± 0.086	1.363** ± 0.264	2.934** ± 1.046	0.337** ± 0.251	0.803** ± 0.163	1.371** ± 0.473	0.662** ± 0.027
	PLIP [18]	1.618** ± 0.309	3.862** ± 1.126	0.260** ± 0.075	1.364** ± 0.248	2.832** ± 0.813	0.350** ± 0.061	0.801** ± 0.173	1.364** ± 0.442	0.658** ± 0.041
	PLIP† [18]	1.921** ± 0.544	4.168** ± 2.409	0.203** ± 0.142	1.497** ± 0.485	3.400** ± 1.875	0.303** ± 0.154	0.875** ± 0.251	1.545** ± 0.872	0.609** ± 0.068
	UNI [16]	1.702** ± 0.392	4.275** ± 1.551	0.211** ± 0.078	1.433** ± 0.303	3.142** ± 1.097	0.305** ± 0.072	0.849** ± 0.188	1.496** ± 0.546	0.622** ± 0.037
	UNI† [16]	1.641** ± 0.293	4.056** ± 1.109	0.199** ± 0.075	1.439** ± 0.168	3.226** ± 0.555	0.302** ± 0.066	0.868** ± 0.096	1.528** ± 0.238	0.607** ± 0.050
	Gigapath [17]	1.588** ± 0.294	3.694** ± 0.934	0.242** ± 0.110	1.334** ± 0.228	2.705** ± 0.692	0.343** ± 0.083	0.818** ± 0.155	1.347** ± 0.391	0.643** ± 0.062
	Gigapath† [17]	1.617** ± 0.117	3.987** ± 0.293	0.201** ± 0.117	1.374** ± 0.044	3.011** ± 0.034	0.309** ± 0.102	0.840** ± 0.120	1.517** ± 0.300	0.615** ± 0.035
	Ours	**1.471 ± 0.220**	**3.094 ± 0.731**	**0.315 ± 0.110**	**1.249 ± 0.136**	**2.390 ± 0.333**	**0.404 ± 0.083**	**0.759 ± 0.130**	**1.158 ± 0.293**	**0.701 ± 0.043**
CrunchDAO-Bowel	ST-Net [12]	0.240** ± 0.015	0.098** ± 0.011	0.659** ± 0.039	0.210** ± 0.014	0.079 ± 0.009	0.654** ± 0.035	0.146* ± 0.011	0.049* ± 0.006	0.661** ± 0.030
	DeepSpaCE [13]	0.252** ± 0.015	0.106** ± 0.012	0.622** ± 0.041	0.214* ± 0.015	0.083 ± 0.009	0.631** ± 0.038	0.138 ± 0.010	0.049* ± 0.006	0.663** ± 0.026
	HisToGene [14]	0.263** ± 0.023	0.113* ± 0.014	0.587** ± 0.056	0.233** ± 0.010	0.091 ± 0.009	0.582** ± 0.029	0.156** ± 0.008	0.054* ± 0.005	0.613** ± 0.022
	BLEEP [20]	0.228 ± 0.018	0.101 ± 0.013	0.661** ± 0.041	0.196 ± 0.020	0.081 ± 0.013	0.658** ± 0.037	**0.124 ± 0.011**	0.046** ± 0.005	0.692 ± 0.027
	Loki [51]	0.231 ± 0.015	0.094 ± 0.015	0.690 ± 0.034	0.201 ± 0.012	0.074 ± 0.010	0.692 ± 0.025	0.139 ± 0.008	0.045 ± 0.006	0.691 ± 0.010
	STPath [52]	0.223 ± 0.019	0.089 ± 0.008	0.700 ± 0.062	0.200 ± 0.015	0.074 ± 0.012	0.694 ± 0.031	0.133 ± 0.011	0.044 ± 0.004	0.707 ± 0.023
	PLIP [18]	0.231 ± 0.013	0.092 ± 0.010	0.683 ± 0.037	0.201 ± 0.013	0.073 ± 0.009	0.683 ± 0.034	0.132 ± 0.009	0.043 ± 0.006	0.704 ± 0.027
	PLIP† [18]	0.231 ± 0.015	0.090 ± 0.011	0.699 ± 0.036	0.195 ± 0.013	0.072 ± 0.007	0.698 ± 0.027	0.131 ± 0.010	0.043 ± 0.004	0.711 ± 0.020
	UNI [16]	0.228 ± 0.011	0.089 ± 0.007	0.701 ± 0.028	0.200 ± 0.012	0.071 ± 0.008	0.699 ± 0.027	0.138 ± 0.009	0.043 ± 0.005	0.710 ± 0.024
	UNI† [16]	0.223 ± 0.016	0.091 ± 0.011	0.700 ± 0.030	0.199 ± 0.017	0.078 ± 0.010	0.677 ± 0.033	0.135 ± 0.014	0.047 ± 0.007	0.686 ± 0.041
	Gigapath [17]	0.229 ± 0.011	0.090 ± 0.007	0.696 ± 0.026	0.202 ± 0.011	0.073 ± 0.006	0.689 ± 0.028	0.141 ± 0.009	0.044 ± 0.004	0.703 ± 0.024
	Gigapath† [17]	0.225 ± 0.009	0.090 ± 0.005	0.700 ± 0.011	0.196 ± 0.011	0.072 ± 0.006	0.700 ± 0.025	0.140 ± 0.011	0.047 ± 0.004	0.707 ± 0.022
	Ours	**0.223 ± 0.014**	**0.087 ± 0.009**	**0.701 ± 0.028**	**0.193 ± 0.015**	**0.071 ± 0.008**	**0.701 ± 0.028**	0.129 ± 0.011	**0.042 ± 0.005**	**0.715 ± 0.026**

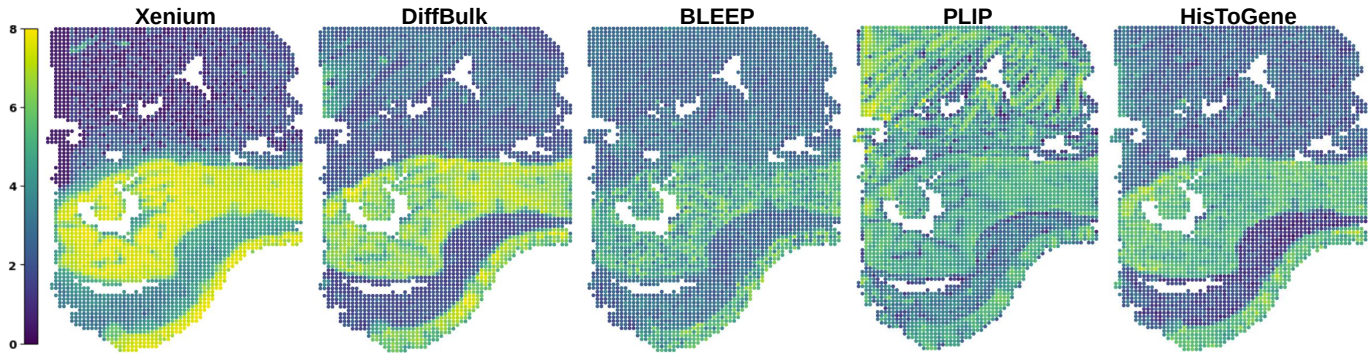

Fig. 2. Spatial expression profile of the gene CD24 in a human bowel tissue section. Each dot corresponds to a ST capture spot, color-coded by the normalized expression level of CD24 (purple: low, yellow: high).

ability to preserve the permutation-invariance property of gene modality.

To further address potential bias caused by limited data scale, we additionally train an *MLP-based (multiple datasets)* variant, where the input dimension is defined as the union of all genes across datasets and the missing genes for each dataset are zero-padded. This variant slightly improves over the single-dataset MLP baseline (MAE: 0.927 → 0.909, PCC: 0.434 → 0.442), suggesting that access to larger and more diverse data indeed benefits conventional MLPs. However, it still falls short of our proposed open-embedding encoder, which achieves 0.864 MAE and 0.458 PCC when trained jointly on all datasets.

These results highlight that although increasing the data scale benefits conventional MLPs, its lack of permutation invariance limits its ability to effectively model heterogeneous

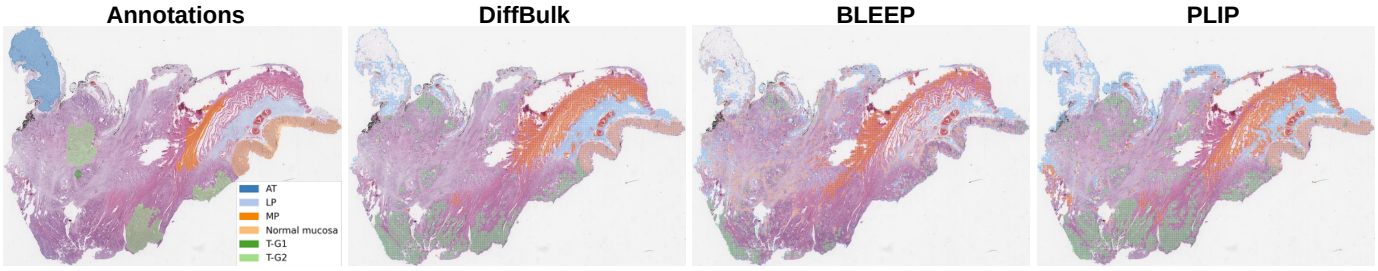

Fig. 3. Visualization of tissue architecture based on gene expression clustering. The left panel shows manual annotations by a pathologist, while the remaining panels display unsupervised clustering results derived from gene expression similarity.

TABLE II
QUANTITATIVE RESULTS OF THE ABLATION STUDY ANALYZING THE IMPACT OF DIFFERENT COMPONENTS IN OUR FRAMEWORK.

Configuration	MAE ↓	MSE ↓	PCC ↑
Baseline (only the second stage)	1.070	1.905	0.411
+ Diffusion Pretraining	0.898	1.603	0.435
+ Diffusion Pretraining + PMS	0.875	1.583	0.442
+ Diffusion Pretraining + PMS + Noise	**0.864**	**1.551**	**0.458**

TABLE III
ABLATION RESULTS COMPARING DIFFERENT GENE ENCODER ARCHITECTURES.

Gene Encoder	MAE ↓	MSE ↓	PCC ↑
MLP-based (single dataset)	0.927	1.737	0.434
MLP-based (multiple datasets)	0.909	1.626	0.442
Open-Embedding (single dataset)	0.912	1.663	0.457
Open-Embedding (multiple datasets)	**0.864**	**1.551**	**0.458**

TABLE IV
EFFECT OF THE NUMBER OF ENCODER BLOCKS IN THE OPEN-EMBEDDING GENE ENCODER.

# of Blocks	MAE ↓	MSE ↓	PCC ↑
1	0.984	1.893	0.422
2	0.864	1.551	0.458
4	**0.841**	**1.469**	**0.467**

TABLE V
ABLATION STUDY ON DIFFERENT LAYERS IN U-NET.

Layers	MAE ↓	MSE ↓	PCC ↑
All	0.880	1.602	0.435
Encoder	0.880	1.596	0.433
Decoder	**0.864**	**1.551**	**0.458**

gene panels. In contrast, our open-embedding gene encoder is explicitly designed to handle variable gene sets and preserve permutation invariance property of gene expression data, enabling it to capture meaningful cross-gene relationships. This demonstrates that the performance improvement stems from the model design itself, rather than merely from increased data scale, thereby validating the robustness and necessity of the proposed open-embedding gene encoder.

We further investigate the impact of the number of encoder blocks in the open-embedding gene encoder. As shown in Table IV, increasing the number of transformer blocks consistently improves performance across all metrics, particularly from one to two blocks (MAE: 0.984 → 0.864, PCC: 0.422 → 0.458). When further increasing to four blocks, the performance gain becomes marginal (MAE: 0.864 → 0.841, PCC: 0.458 → 0.467), suggesting that the model begins to saturate and additional depth offers diminishing returns. This observation confirms that our current two-block configuration provides an effective trade-off between representational capacity and generalization. It demonstrates that the encoder is well-aligned with the available data scale and not underutilized.

F. Effectiveness of Different Layers in U-Net

Different layers of the pretrained U-Net capture features at varying levels of abstraction, and these representations can have a significant impact on the downstream pseudo-bulk gene expression prediction. To identify the most effective part for this task, we evaluate three configurations: using activations from (1) the encoder only, (2) the decoder only, and (3) the entire U-Net.

The U-Net encoder progressively downsamples the input to capture local texture, morphology, and mid-level patterns, while the decoder upsamples from low resolution, aggregating global context and high-level information via skip connections. As shown in Table V, the decoder-only branch outperforms both encoder-only and full U-Net, which we attribute to its favorable balance of semantic richness and spatial resolution: during gene-conditional diffusion training the decoder is directly optimized to reconstruct noise-free images conditioned on gene expression, so gene-relevant semantics become more explicitly embedded, whereas the encoder tends to emphasize local visual cues and the full U-Net can introduce redundant or noisy signals that dilute those gene-specific features. The skip connections further help the decoder fuse low- and high-level information, enhancing its ability to represent spatially distributed expression patterns. These findings underscore that selecting feature layers with strong semantic alignment to the downstream task improves the effectiveness of pretrained diffusion models.

G. Effectiveness of the Probability p in PMS

We investigate the impact of the probability p in gene-to-image conditional diffusion training. We examine three representative values of p. When $p = 0$, the U-Net is conditioned on gene expression, following a fully conditional training

TABLE VI
ABLATION STUDY ON THE PROBABILITY p IN PMS AND THE NOISE LEVEL σ.

p	σ	MAE ↓	MSE ↓	PCC ↑
0.5	1.000	0.929	1.741	0.456
0.5	0.100	0.875	1.611	0.456
0.5	0.015	0.870	1.570	0.455
0.5	0.010	**0.864**	**1.551**	**0.458**
0.5	0.001	0.873	1.565	0.458
0.5	0.000	0.875	1.582	0.449
0.0	0.010	0.888	1.616	0.450
1.0	0.010	0.895	1.662	0.445

TABLE VII
ABLATION STUDY ON THE FOUNDATION MODEL BRANCH. BLUE NUMBERS DENOTE THE IMPROVEMENT OF EACH COMBINED MODEL COMPARED TO ITS CORRESPONDING FM-ALONE BASELINE.

Model	MAE ↓	MSE ↓	PCC ↑
BLEEP	1.067	2.228	0.414
Diffusion (alone)	0.964	1.718	0.433
Gigapath+Diffusion	1.115-0.129	1.962-0.217	0.416+0.031
PLIP+Diffusion	**1.135-0.271**	**1.938-0.387**	0.433+0.025
UNI+Diffusion	1.146-0.158	2.020-0.240	0.407+0.036
UNI2+Diffusion	1.092-0.174	1.773-0.162	0.431+0.031
Virchow2+Diffusion	1.169-0.183	2.022-0.211	0.401+0.038
H-optimus+Diffusion	1.212-0.186	1.801-0.151	**0.418+0.048**

regime. When $p = 1$, the U-Net is never conditioned on gene expression, reducing the training to a standard diffusion process. The intermediate setting $p = 0.5$ randomly conditions the U-Net on gene expression in 50% of the training steps.

As shown in Table VI, the best performance is achieved when $p = 0.5$. This balanced configuration enables the model to remain responsive to gene-specific guidance while also learning to operate in gene-agnostic settings in downstream tasks. In contrast, setting $p = 0$ leads the model to over-rely on gene inputs during pretraining, degrading performance when gene data is absent at inference. On the other hand, $p = 1$ removes all gene guidance, preventing the model from learning gene-relevant features. These findings suggest that $p = 0.5$ effectively strikes an effective trade-off, enabling the model to develop robust and transferable representations.

H. Influence of Different Noise Levels

We observed that our framework achieves better performance when low-level noise $\mathcal{N}(0, \sigma^2 I)$ is added to pathology images during the downstream stage. We conducted experiments with varying levels of noise intensity σ. As shown in Table VI, the model performs optimally when σ lies within the range $0 < \sigma \leq 0.01$. This performance gain can be attributed to the nature of the conditional diffusion training, where the U-Net is optimized to reconstruct clean signals from noisy inputs. Since the pretrained U-Net is rarely exposed to noise-free histological images during training, introducing a moderate level of noise at the representation learning stage creates a consistent training distribution that aligns with its denoising objective. However, when the noise level becomes too high (i.e., $\sigma > 0.01$), the injected perturbations start to obscure critical histological features necessary for gene expression prediction. These findings highlight the importance of carefully tuning the noise level to match the denoising nature of the conditional diffusion process while preserving the fidelity of informative image features.

I. Effectiveness of the Foundation Model Branch

We first ablate the FM branch to assess the standalone effectiveness of the conditional diffusion-based image features for gene expression prediction. As shown in Table VII, *Diffusion(alone)* refers to the model that utilizes only the diffusion-based features. Notably, this model achieves performance comparable to BLEEP, which validates the effectiveness of our gene-to-image conditional diffusion training framework and open-embedding gene encoder. This demonstrates that the pretrained U-Net can extract gene-relevant visual representations without the help of additional pretrained FM encoders.

To further investigate whether incorporating an additional FM branch can enhance performance, we combine diffusion-based features with a variety of FM backbones, including PLIP, UNI, Gigapath, Virchow 2 [54], H-optimus [55], and UNI-2 [56]). Each *+Diffusion* variant corresponds to a combination of diffusion features with a specific FM backbone. Across all FM backbones, integrating diffusion-based features consistently improves performance compared with their FM-alone counterparts, as highlighted by the blue numbers in Table VII. This demonstrates that the gene-specific priors learned through conditional diffusion training provide complementary information to foundation model embeddings, regardless of the FM's scale or training paradigm.

Interestingly, the magnitude of improvement varies with the FM choice. Among models, PLIP+Diffusion achieves the largest gain, which we attribute to PLIP's multimodal contrastive pretraining that aligns histology–text pairs and thus benefits from the diffusion branch's gene-conditioned visual cues. For stronger FMs such as UNI-2, Virchow2, and H-optimus, the diffusion-enhanced variants still achieve noticeable improvements (+0.031–0.048 in PCC), indicating that even highly expressive visual encoders can benefit from the biologically grounded structure provided by the diffusion pretraining. This confirms the generality and compatibility of DiffBulk with next-generation pathology foundation models, highlighting its potential to enhance existing FMs through gene-aware structural conditioning.

V. Discussion and Conclusion

Predicting gene expression from histopathological images offers a cost-effective alternative to ST, which remains expensive and labor-intensive. However, existing approaches typically cast this task as a direct regression problem, ignoring its inherently ill-posed nature and relying heavily on MLP-based gene encoders that are limited in scalability and violate permutation invariance. These constraints often result in suboptimal performance. To address these challenges, we employ a gene-to-image conditional diffusion model to implicitly model the

complex relationship between gene expression profiles and histological features. Our design introduces an open-embedding gene encoder that respects the permutation-invariant structure of gene data and supports flexible integration across heterogeneous gene sets. To ensure the utility of learned diffusion representations during downstream prediction, we further introduce a PMS module. Although the diffusion model is capable of generating pathological image patches conditioned on gene expression, we emphasize that image synthesis fidelity is not the primary objective of DiffBulk; rather, the diffusion process is leveraged as a powerful pretraining mechanism to encode subtle gene-dependent morphological cues into image-perceptual representations.

In the downstream stage, we extract multi-scale features from the frozen conditional diffusion model and integrate them with features from a FM via a gated fusion module. We further experiment with alternative fusion mechanisms, including (1) a learnable additive fusion defined as $\mathbf{z} = c_{\text{add}} \cdot \mathbf{z}_{\text{fm}} + (1 - c_{\text{add}}) \cdot \mathbf{z}_{\text{diff}}$, and (2) a concatenation-based fusion defined as $\mathbf{z} = \text{Concat}(\mathbf{z}_{\text{fm}}, c_{\text{concat}} \cdot \mathbf{z}_{\text{diff}})$, where c_{add} and c_{concat} are learned scalars. Although these variants underperform compared to our gated design, the learned weights ($c_{\text{add}} = 0.7088$, $c_{\text{concat}} = 0.7348$) indicate that FM-based features currently dominate prediction, highlighting the need for more discriminative and expressive diffusion-based representations. This insight highlights a promising direction for future research: the development of stronger conditional pretraining objectives and more powerful diffusion-based feature extractors.

While DiffBulk demonstrates strong predictive performance and generalizability across multiple datasets, several limitations remain. First, DiffBulk is designed as a two-stage framework rather than a fully end-to-end architecture. It may limit the joint fine-tuning of gene-conditional and predictive modules. In future work, an end-to-end optimization scheme could be explored to further improve representation alignment and reduce training complexity.

Second, the introduction of the diffusion module inevitably increases computational cost. The diffusion U-Net adds more parameters and FLOPs compared with task-specific baselines, leading to higher training time and GPU memory consumption. Although the diffusion backbone is frozen during inference—mitigating runtime overhead—this additional cost may hinder large-scale deployment or real-time applications. Designing lightweight diffusion architectures or knowledge distillation strategies could help alleviate this issue.

In conclusion, DiffBulk introduces a principled and extensible framework for learning gene-aware image representations via conditional diffusion modeling. Our permutation-invariant, open-embedding gene encoder enables scalable pretraining across diverse ST datasets, while our gated fusion design effectively combines complementary information from foundation models. Extensive evaluations on three Xenium ST datasets demonstrate the robustness, generalizability, and performance superiority of DiffBulk over existing methods. Beyond gene expression prediction, our framework opens up new possibilities for integrating generative pretraining into spatial omics analysis and computational pathology.

REFERENCES

[1] F. Aeffner *et al.*, "Introduction to digital image analysis in whole-slide imaging: a white paper from the digital pathology association," *Journal of pathology informatics*, vol. 10, no. 1, p. 9, 2019.

[2] P. Wirapati *et al.*, "Meta-analysis of gene expression profiles in breast cancer: toward a unified understanding of breast cancer subtyping and prognosis signatures," *Breast Cancer Research*, vol. 10, pp. 1–11, 2008.

[3] M. Garg, "Rna sequencing: A revolutionary tool for transcriptomics," in *Advances in Animal Genomics*. Elsevier, 2021, pp. 61–73.

[4] Z. Wang, M. Gerstein, and M. Snyder, "Rna-seq: a revolutionary tool for transcriptomics," *Nature reviews genetics*, vol. 10, no. 1, pp. 57–63, 2009.

[5] M. J. Stubbington, O. Rozenblatt-Rosen, A. Regev, and S. A. Teichmann, "Single-cell transcriptomics to explore the immune system in health and disease," *Science*, vol. 358, no. 6359, pp. 58–63, 2017.

[6] G. X. Zheng *et al.*, "Massively parallel digital transcriptional profiling of single cells," *Nature communications*, vol. 8, no. 1, p. 14049, 2017.

[7] S. C. van den Brink *et al.*, "Single-cell sequencing reveals dissociation-induced gene expression in tissue subpopulations," *Nature methods*, vol. 14, no. 10, pp. 935–936, 2017.

[8] P. L. Ståhl *et al.*, "Visualization and analysis of gene expression in tissue sections by spatial transcriptomics," *Science*, vol. 353, no. 6294, pp. 78–82, 2016.

[9] M. Kiuru *et al.*, "High-plex spatial rna profiling reveals cell type–specific biomarker expression during melanoma development," *Journal of Investigative Dermatology*, vol. 142, no. 5, pp. 1401–1412, 2022.

[10] F. Avila Cobos, J. Alquicira-Hernandez, J. E. Powell, P. Mestdagh, and K. De Preter, "Benchmarking of cell type deconvolution pipelines for transcriptomics data," *Nature communications*, vol. 11, no. 1, p. 5650, 2020.

[11] H. L. Crowell *et al.*, "Muscat detects subpopulation-specific state transitions from multi-sample multi-condition single-cell transcriptomics data," *Nature communications*, vol. 11, no. 1, p. 6077, 2020.

[12] B. He *et al.*, "Integrating spatial gene expression and breast tumour morphology via deep learning," *Nature biomedical engineering*, vol. 4, no. 8, pp. 827–834, 2020.

[13] T. Monjo, M. Koido, S. Nagasawa, Y. Suzuki, and Y. Kamatani, "Efficient prediction of a spatial transcriptomics profile better characterizes breast cancer tissue sections without costly experimentation," *Scientific reports*, vol. 12, no. 1, p. 4133, 2022.

[14] M. Pang, K. Su, and M. Li, "Leveraging information in spatial transcriptomics to predict super-resolution gene expression from histology images in tumors," *bioRxiv*, 2021. [Online]. Available: https://www.biorxiv.org/content/early/2021/11/28/2021.11.28.470212

[15] Y. Zeng *et al.*, "Spatial transcriptomics prediction from histology jointly through transformer and graph neural networks," *bioRxiv*, 2022. [Online]. Available: https://www.biorxiv.org/content/early/2022/04/26/2022.04.25.489397

[16] R. J. Chen *et al.*, "Towards a general-purpose foundation model for computational pathology," *Nature Medicine*, vol. 30, no. 3, pp. 850–862, 2024.

[17] H. Xu *et al.*, "A whole-slide foundation model for digital pathology from real-world data," *Nature*, pp. 1–8, 2024.

[18] Z. Huang, F. Bianchi, M. Yuksekgonul, T. J. Montine, and J. Zou, "A visual–language foundation model for pathology image analysis using medical twitter," *Nature medicine*, vol. 29, no. 9, pp. 2307–2316, 2023.

[19] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu, "Coca: Contrastive captioners are image-text foundation models," *arXiv preprint arXiv:2205.01917*, 2022.

[20] R. Xie *et al.*, "Spatially resolved gene expression prediction from histology images via bi-modal contrastive learning," in *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36. Curran Associates, Inc., 2023, pp. 70 626–70 637.

[21] W. Xiang, H. Yang, D. Huang, and Y. Wang, "Denoising diffusion autoencoders are unified self-supervised learners," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 15 802–15 812.

[22] A. C. Li, M. Prabhudesai, S. Duggal, E. Brown, and D. Pathak, "Your diffusion model is secretly a zero-shot classifier," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 2206–2217.

[23] S. Mukhopadhyay *et al.*, "Diffusion models beat gans on image classification," *arXiv preprint arXiv:2307.08702*, 2023.

[24] X. Yang and X. Wang, "Diffusion model as representation learner," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 18 938–18 949.

[25] J. Xu, S. Liu, A. Vahdat, W. Byeon, X. Wang, and S. De Mello, "Open-vocabulary panoptic segmentation with text-to-image diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 2955–2966.

[26] D. Baranchuk, I. Rubachev, A. Voynov, V. Khrulkov, and A. Babenko, "Label-efficient semantic segmentation with diffusion models," *arXiv preprint arXiv:2112.03126*, 2021.

[27] M. Zhang, G. Song, X. Shi, Y. Liu, and H. Li, "Three things we need to know about transferring stable diffusion to visual dense prediction tasks," in *European Conference on Computer Vision*. Springer, 2024, pp. 128–145.

[28] J. Tian, F. Yu *et al.*, "Diffuse, attend, and segment: Unsupervised zero-shot segmentation using stable diffusion," in *CVPR*, 2024. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2024/papers/Tian_Diffuse_Attend_and_Segment_Unsupervised_Zero-Shot_Segmentation_using_Stable_Diffusion_CVPR_2024_paper.pdf

[29] J. Zhang *et al.*, "A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence," *Advances in Neural Information Processing Systems*, vol. 36, pp. 45 533–45 547, 2023.

[30] G. Luo, L. Dunlap, D. H. Park, A. Holynski, and T. Darrell, "Diffusion hyperfeatures: Searching through time and space for semantic correspondence," *Advances in Neural Information Processing Systems*, vol. 36, pp. 47 500–47 510, 2023.

[31] E. Hedlin *et al.*, "Unsupervised semantic correspondence using stable diffusion," *Advances in Neural Information Processing Systems*, vol. 36, pp. 8266–8279, 2023.

[32] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[33] Y. Song and S. Ermon, "Generative modeling by estimating gradients of the data distribution," *Advances in neural information processing systems*, vol. 32, 2019.

[34] G. Jaume *et al.*, "Hest-1k: A dataset for spatial transcriptomics and histology image analysis," *arXiv preprint arXiv:2406.16192*, 2024.

[35] CrunchDAO, "Broad-1: Crunchdao competition dataset," https://hub.crunchdao.com/competitions/broad-1, 2024, accessed May 23, 2025.

[36] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[37] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[38] A. Dosovitskiy, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[39] S. Zhu, Y. Zhu, M. Tao, and P. Qiu, "Diffusion generative modeling for spatially resolved gene expression inference from histology images," *arXiv preprint arXiv:2501.15598*, 2025.

[40] M. Oquab *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[41] T. Lin, Z. Chen, Z. Yan, W. Yu, and F. Zheng, "Stable diffusion segmentation for biomedical images with single-step reverse process," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2024, pp. 656–666.

[42] Z. Xiao *et al.*, "Diffusion model based on generalized map for accelerated mri," *NMR in Biomedicine*, vol. 37, no. 12, p. e5232, 2024.

[43] P. Tan, M. Geng, J. Lu, L. Shi, B. Huang, and Q. Liu, "Msdiff: Multi-scale diffusion model for ultra-sparse view ct reconstruction," *arXiv preprint arXiv:2405.05814*, 2024.

[44] A. Radford *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[45] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 16 000–16 009.

[46] A. Vaswani *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[47] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.

[48] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[49] D. P. Kingma, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[50] I. Loshchilov, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[51] W. Chen *et al.*, "A visual–omics foundation model to bridge histopathology with spatial transcriptomics," *Nature Methods*, pp. 1–15, 2025.

[52] T. Huang, T. Liu, M. Babadi, R. Ying, and W. Jin, "Stpath: A generative foundation model for integrating spatial transcriptomics and whole slide images," *bioRxiv*, pp. 2025–04, 2025.

[53] E. Sagiv *et al.*, "Cd24 is a new oncogene, early at the multistep process of colorectal cancer carcinogenesis," *Gastroenterology*, vol. 131, no. 2, pp. 630–639, 2006.

[54] E. Zimmermann *et al.*, "Virchow2: Scaling self-supervised mixed magnification models in pathology," *arXiv preprint arXiv:2408.00738*, 2024.

[55] C. Saillard *et al.*, "H-optimus-0," 2024. [Online]. Available: https://github.com/bioptimus/releases/tree/main/models/h-optimus/v0

[56] R. J. Chen *et al.*, "Towards a general-purpose foundation model for computational pathology," *Nature Medicine*, 2024.