HyBoxST: Coupling Hyperbolic Spatial Representations with Molecular Program Boxes for Transcriptomics Prediction from Histology

Anonymous authors
Paper under double-blind review

Abstract

Predicting spatial gene expression from histology provides a scalable way to profile tissue molecular states. Although existing methods increasingly capture tissue morphology and spatial context, they rarely model the pathway structure among predicted genes explicitly. This leaves known relationships among functionally related genes underused when learning their expression from tissue morphology. This makes it difficult for these models to accurately recover coordinated expression changes among genes. Here we propose **HyBoxST**, which incorporates gene-to-pathway memberships into expression prediction through a pathway-guided structured decoder. HyBoxST represents pathways as learnable boxes and genes as points in a shared molecular latent space, so that pathway membership can be expressed through containment and shared membership through overlap between pathway boxes. For each spot, features from the spot image and its surrounding niche generate a query box. Its overlap with pathway boxes is used to adjust directly predicted pathway expression scores according to the local tissue context. For each gene, the decoder aggregates the corrected scores from pathways to which the gene belongs, while a residual branch captures expression variation beyond the selected pathways. Across four tissue datasets, HyBoxST consistently improves gene expression prediction. Additional controls with randomized gene-pathway memberships and spatial analyses of pathway signals show that the improvement relies on the biological pathway structure rather than the mask structure alone.

1 Introduction

Spatial transcriptomics (ST) measures gene expression while preserving spatial information within tissue, providing a means to characterize tumor regions, immune infiltration, and stromal architecture (Stahl et al., 2016; Williams et al., 2022; Zhang et al., 2022). Studies of breast and colorectal cancer have further revealed potential links among tissue architecture, genomic features, and treatment response (Jiménez-Santos et al., 2024; Choo et al., 2026). However, the high cost of ST experiments and demanding requirements for sample preparation and tissue handling limit their application to large cohorts, particularly retrospective clinical cohorts. In contrast, hematoxylin and eosin (H&E) slides are routinely used in diagnostic pathology, and substantial collections of archival specimens are available. Predicting spatial gene expression from H&E images therefore offers a promising route to obtaining spatial molecular information from existing pathology data (He et al., 2020; Monjo et al., 2022).

Recent studies have developed methods for mapping H&E images to spatial gene expression through visual representations at multiple scales (Chung et al., 2024), spatial modeling (Zhang et al., 2026), contrastive learning and retrieval augmentation (Xie et al., 2023), and generative modeling (Zhu et al., 2025; Huang et al., 2025). These advances demonstrate that tissue morphology contains rich molecular information. However, most models still primarily predict gene expression directly from image representations and make limited use of the functional organization of genes. Because local tissue morphology is often associated with molecular programs involving multiple genes, pathways

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

provide a natural intermediate structure linking morphology to gene expression prediction (Jaume et al., 2024b; Majumder et al., 2026). Existing methods introduce functional priors through pathway alignment, gene graphs (Shi et al., 2025), or hierarchical decoding (Section 2.2). Building on these developments, we ask how to represent gene membership in overlapping pathways and use local image context to estimate pathway signals that guide predictions of their member genes.

Pathways form overlapping gene sets: a gene can participate in multiple pathways (Gillespie et al., 2022). Learning a separate vector for each pathway does not directly specify how shared membership should be reflected in the representation space. Box embeddings provide a natural geometric representation for this purpose (Vilnis et al., 2018; Ren et al., 2020; Dasgupta et al., 2020). Pathways are represented as regions in a latent space and genes as points; containment of a gene point within a pathway box expresses membership, while intersections between boxes accommodate shared genes. This representation explicitly encodes pathway overlap and allows local image representations to query pathway signals that subsequently constrain expression predictions for member genes.

We introduce **HyBoxST**, a method for predicting spatial gene expression from histology using known pathway memberships. We obtain pathway gene sets from Reactome and Hallmark (Liberzon et al., 2015) and restrict them to the selected prediction genes. Building on box embeddings for knowledge representation and query reasoning (Vilnis et al., 2018; Ren et al., 2020), we represent pathways as learnable boxes and genes as points in a shared latent space. The annotated memberships encourage gene points to lie within their pathway boxes, allowing shared genes to occupy intersections. For each spot, features from its image and the surrounding niche generate a *query box*: a region in the same latent space whose location and size depend on the images. Its overlap with each pathway box is transformed into a correction to a pathway expression score predicted directly from the image features. These scores estimate the mean transformed expression of each pathway's measured member genes. A pathway guided decoder then aggregates the corrected scores for each gene only from pathways containing it and combines this estimate with a direct gene prediction. This makes pathway scores a shared intermediate in gene prediction while allowing expression beyond the retained pathway annotations to be modeled.

Our contributions are:

- We introduce HyBoxST, a method that uses pathway expression scores as shared intermediates between histology and gene predictions. Annotated memberships guide how these scores reach individual genes, while direct gene predictions accommodate incomplete pathway coverage.
- We develop an image driven box query mechanism for correcting pathway estimates. Pathway boxes are learned with gene membership supervision, and their overlap with a query box generated from local images supplies a correction for each pathway.
- Experiments on four tissue datasets evaluate gene prediction, while controlled comparisons support contributions from box queries, membership supervision, and biological pathway annotations. Analyses on Kidney show how query corrections change pathway estimates across tissue locations.

2 Related Work

2.1 Histology based spatial transcriptomics prediction

Existing work primarily improves histology to expression prediction through visual representation, spatial context modeling, and cross-modal alignment. StNet predicts each spot's expression profile primarily from a local image patch, limiting its access to tissue context beyond local morphology (He et al., 2020). TRIPLEX combines spot, neighborhood, and global visual features to broaden the tissue context used for prediction (Chung et al., 2024); however, richer visual information does not directly constrain expression relationships among functionally related genes, and lower errors for individual genes do not guarantee recovery of their coordinated changes. BLEEP retrieves reference expression profiles through cross-modal alignment to capture gene associations from existing expression patterns (Xie et al., 2023), but averaging multiple reference profiles may smooth genuine local biological variation, making abrupt expression signals difficult to preserve. These limitations point to a common concern: numerical prediction accuracy does not necessarily imply preserva-

tion of biological expression structure. From an evaluation perspective, HistoPrism emphasizes that correlation on highly variable genes alone is insufficient to assess coordinated expression among functionally related genes, and introduces pathway coherence as a complementary metric (Hu et al., 2026). This further motivates considering functional relationships among genes alongside individual gene accuracy.

2.2 Pathway guided and functionally coherent prediction

To incorporate such functional relationships into learning, PEaRL aligns histology with pathway representations derived from ssGSEA (Majumder et al., 2026). However, it subsequently freezes the backbone and trains separate prediction heads, so the gene prediction objective cannot further adapt the shared representation. MSGR incorporates a Gene Ontology (GO) hierarchy into expression decoding with residual refinement (Xu et al., 2026). Because GO permits multiple parents, its default tree projection discards some original relationships. Its variant that restores multiple gene memberships uniformly averages parent predictions, assigning equal contributions to different functional sources. This variant yields lower PCC than the default tree on all nine tested datasets, showing that restoring memberships through uniform averaging alone does not improve prediction in this setting. An open challenge is therefore to preserve overlapping functional memberships while allowing the contributions of functional groups to gene prediction to reflect local tissue context.

2.3 Geometric representations of membership and overlap

From a representation learning perspective, this challenge involves modeling overlapping sets and their relationships. Representing a set as a single point does not explicitly describe its boundaries or intersections. Query2box represents query answer sets as continuous regions for knowledge graph reasoning, using box geometry to model membership and intersection (Ren et al., 2020). This regional representation provides an inductive bias for functional groups with shared gene members. Building on this idea, we represent pathways as learnable regions that can overlap and generate a query from spot and surrounding niche images. Query overlap with pathway regions corrects predicted pathway expression scores, which a learned mapping constrained by memberships passes to genes. This preserves overlapping memberships and allows pathway contributions to vary with local tissue context through the corrected scores.

3 Method

HyBoxST incorporates overlapping pathways into expression prediction by representing them as shared molecular program boxes. Figure 1 connects these global pathway regions to the prediction for an individual spot. Spot and niche images provide a joint representation h_i, from which the model predicts pathway expression directly and generates a query box. Matching the query with the shared boxes produces a correction to the direct pathway estimates. The corrected scores are aggregated for member genes and combined with a direct gene prediction. The upper and lower blocks distinguish training supervision from the image based inference procedure.

3.1 Problem setup and spatial representation

Given a spot image X_i^s and its surrounding niche image X_i^n, our goal is to predict the expression vector $Y_i \in \mathbb{R}^G$ of G selected genes, with $Y_{ig} = \log_2(1 + x_{ig})$ for raw count x_{ig}. We use a fixed membership matrix $M \in \{0,1\}^{P\times G}$, where $M_{pg} = 1$ indicates that gene g belongs to pathway p. The pathways are drawn from Reactome and Hallmark and restricted to the selected gene panel (Appendix C.1).

Local morphology and its tissue context provide complementary evidence for molecular prediction. A shared pathology encoder E_{img}, followed by learned projections ϕ_s and ϕ_n, produces their joint representation:

$$h_i = [\tilde{I}_i^s; \tilde{I}_i^n], \qquad \tilde{I}_i^s = \phi_s(E_{\text{img}}(X_i^s)), \quad \tilde{I}_i^n = \phi_n(E_{\text{img}}(X_i^n)). \tag{1}$$

The representation h_i supplies the query, direct pathway, and direct gene heads below. During training, we follow the spatial alignment framework of Zhang et al. (2026) to align image and expression

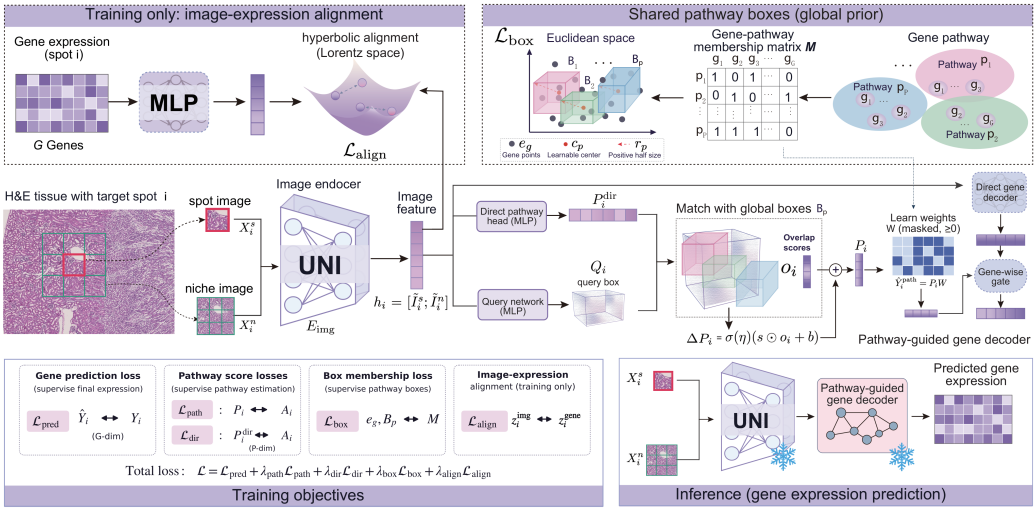

Figure 1: Overview of HyBoxST. Top: image and expression representations are aligned during training, while the fixed membership matrix M supervises gene points and shared pathway boxes in a separate Euclidean space. Middle: the UNI encoder extracts spot and niche image features to form h_i. A direct pathway head predicts P_i^{dir}, and a query network generates Q_i. Matching Q_i with the pathway boxes B_p yields overlap scores o_i, which are transformed into ΔP_i and added to P_i^{dir} to obtain P_i. Nonnegative weights W, constrained by M, aggregate P_i into a gene prediction, which a gene gate combines with the direct gene branch. Bottom left: gene prediction, pathway prediction, box membership, and alignment losses jointly train the model. Bottom right: inference uses paired images and the trained model with fixed parameters, without observed expression.

representations at the spot and niche levels. This supervision encourages molecularly informative image features; the decoder uses h_i before the Lorentz mapping. The alignment objective $\mathcal{L}_{\text{align}}$ and encoder details are given in Appendix A.1.

3.2 Molecular program boxes and image conditioned queries

Pathways share genes, so their representations should accommodate membership in multiple molecular programs. We represent each pathway as an axis aligned box and each gene as a learnable point $e_g \in \mathbb{R}^D$ in a shared Euclidean space (Vilnis et al., 2018; Ren et al., 2020; Dasgupta et al., 2020). A pathway box has learnable center c_p and positive half size r_p:

$$B_p = [c_p - r_p, c_p + r_p], \qquad r_p = \text{softplus}(a_p) + \epsilon_B. \tag{2}$$

The unconstrained parameter a_p determines the half size through softplus; the resulting bounds of B_p are used for both membership supervision and query matching. Membership supervision encourages e_g to lie inside B_p when $M_{pg} = 1$, allowing a shared gene to occupy the intersection of its pathways. We penalize member points outside their boxes and nonmember points within a margin of them using $\mathcal{L}_{\text{box}}$ (Appendix A.2). Because the same gene point participates in all its membership constraints, these constraints jointly shape the pathway regions. Gene points serve only this training role; their influence on expression prediction is mediated by the learned boxes, which provide the reference regions for image queries.

To estimate how local tissue context relates to the pathway regions, we map h_i to a query box in the same D dimensional molecular space (Ren et al., 2020):

$$Q_i = [\mu_i - \delta_i, \mu_i + \delta_i], \qquad (\mu_i, \rho_i) = f_{\text{query}}(h_i), \quad \delta_i = \text{softplus}(\rho_i) + \epsilon_{\text{size}}. \tag{3}$$

The positive constants ϵ_B and ϵ_{size} prevent degenerate boxes. Whereas B_p encodes a pathway shared across spots, Q_i varies with the observed morphology. Their overlap measures how much of the image inferred region matches each pathway. Writing the lower and upper bounds as (l_i^Q, u_i^Q)

and (l_p^B, u_p^B), we use

$$o_{ip} = \exp\left[\mathrm{clip}\left(\frac{1}{D}\sum_{d=1}^{D}\log\frac{\tilde{v}_{ipd}}{\tilde{q}_{id}}, -30, 0\right)\right], \tag{4}$$

where $\tilde{v}_{ipd}$ is the smoothed intersection width, obtained by applying $\mathrm{softplus}_\tau(t) = \tau\log(1+e^{t/\tau})$ to $\min(u_{i,d}^Q, u_{p,d}^B) - \max(l_{i,d}^Q, l_{p,d}^B)$ and adding ϵ. The denominator is the stabilized query width $\tilde{q}_{id} = \max(u_{i,d}^Q - l_{i,d}^Q, \epsilon) + \epsilon$. This geometric mean measures relative overlap across dimensions; smoothing supports matching disjoint regions, and the log formulation avoids multiplying small widths. Numerical properties are detailed in Appendix B.2.

The vector o_i collects the matching scores for all pathways at spot i. It supplies geometric evidence for the correction below, while expression is estimated by a separate prediction head. A direct head predicts $P_i^{\mathrm{dir}} = f_{\mathrm{path}}(h_i)$, and the query contributes a learned correction:

$$P_i = \underbrace{f_{\mathrm{path}}(h_i)}_{P_i^{\mathrm{dir}}} + \underbrace{\sigma(\eta)(s \odot o_i + b)}_{\Delta P_i}. \tag{5}$$

Here $s, b \in \mathbb{R}^P$ and the scalar η are learned, σ is the sigmoid, and $\odot$ denotes elementwise multiplication. Each component of P_i estimates the mean transformed expression of a pathway's measured members. The unconstrained scale s allows a geometric match to increase or decrease this estimate. The resulting P_i supplies the pathway scores to the gene decoder below; P_i^{dir} also receives its own training supervision (Section 3.4). The query network and pathway boxes are optimized jointly through the corrected pathway scores and downstream gene predictions, while $\mathcal{L}_{\mathrm{box}}$ constrains the boxes using annotated memberships. This joint objective learns the relationship between image queries and pathway regions through their contribution to expression prediction (Section 3.4).

3.3 Membership constrained gene decoding

The box query determines the correction to each pathway score; the annotated memberships determine which genes can receive that score. We therefore decode P_i using a separate matrix W of learned nonnegative weights restricted by the same M that supervises the boxes:

$$W_{pg} = \frac{M_{pg}\,\mathrm{softplus}(\omega_{pg})}{\max\left(1, \sum_{p'=1}^{P} M_{p'g}\,\mathrm{softplus}(\omega_{p'g})\right)}, \qquad \hat{Y}_i^{\mathrm{path}} = P_i W. \tag{6}$$

The parameter ω_{pg} controls each annotated pathway's contribution to gene g; these weights are separate parameters, rather than values computed from box overlap. Substituting Equation 5 gives $\hat{Y}_i^{\mathrm{path}} = P_i^{\mathrm{dir}} W + \Delta P_i W$: the query correction reaches member genes through the term $\Delta P_i W$. Each gene can combine corrections from several pathways, with $M_{pg} = 0$ excluding the contribution of pathway p to gene g.

Because the selected pathways capture only part of expression variation, a parallel branch predicts $\hat{Y}_i^{\mathrm{res}} = f_{\mathrm{res}}(h_i)$. The pathway estimate $\hat{Y}_i^{\mathrm{path}}$ and the direct estimate $\hat{Y}_i^{\mathrm{res}}$ are intermediate gene predictions, combined by a learned gene gate:

$$\hat{Y}_{ig}^{\mathrm{raw}} = \gamma_g \hat{Y}_{ig}^{\mathrm{path}} + (1-\gamma_g)\hat{Y}_{ig}^{\mathrm{res}}, \qquad \gamma_g = \mathbf{1}\left[\sum_p M_{pg} > 0\right]\sigma(\theta_g). \tag{7}$$

Covered genes combine pathway and direct predictions; uncovered genes rely entirely on the direct branch. A jointly learned affine layer gives the model output $\hat{Y}_{ig} = \alpha_g \hat{Y}_{ig}^{\mathrm{raw}} + \beta_g$ with $\alpha_g > 0$ (Appendix A.4). The fused estimate $\hat{Y}_i^{\mathrm{raw}}$ is the input to this layer, and $\hat{Y}_i$ denotes the model output supervised by $\mathcal{L}_{\mathrm{pred}}$ during training. This loss optimizes both branches through the model output. At evaluation, a separately fitted gene-wise affine postprocessing map is applied as described in Appendix C.2.

3.4 Joint training objective

We train the model with gene prediction, pathway prediction, membership, and spatial alignment objectives. The gene loss $\mathcal{L}_{\mathrm{pred}}$ combines MSE and MAE with penalties on per gene batch means

and standard deviations (Appendix A.5). To give the intermediate pathway estimates a defined expression scale, we supervise both P_i and P_i^{dir} with the mean expression of measured member genes:

$$\mathcal{L}_{\mathrm{path}} = \mathrm{MSE}_{i,p}(P_{ip}, A_{ip}), \quad \mathcal{L}_{\mathrm{dir}} = \mathrm{MSE}_{i,p}(P_{ip}^{\mathrm{dir}}, A_{ip}), \qquad A_{ip} = \frac{\sum_g M_{pg} Y_{ig}}{\sum_g M_{pg}}. \tag{8}$$

Here $\mathrm{MSE}_{i,p}$ averages squared errors over spots i and pathways p in a mini batch. Each A_{ip} is computed from observed expression to supervise the two pathway estimates during training; it is not an input to the prediction heads. These targets summarize the measured gene panel and are not measurements of whole pathway activity. Together with membership supervision of the boxes and spatial alignment of the representations, the full objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{pred}} + \lambda_{\mathrm{path}}\mathcal{L}_{\mathrm{path}} + \lambda_{\mathrm{dir}}\mathcal{L}_{\mathrm{dir}} + \lambda_{\mathrm{box}}\mathcal{L}_{\mathrm{box}} + \lambda_{\mathrm{align}}\mathcal{L}_{\mathrm{align}}. \tag{9}$$

The coefficients λ balance the training objectives, and backpropagation of $\mathcal{L}$ jointly updates the model parameters. At inference, expression supervision is removed and the model predicts from the paired images using the learned pathway boxes and decoder. Algorithm 1 provides the training procedure; computational complexity is given in Appendix B.7.

4 Experiments

We first evaluate gene prediction across four tissues, then use component ablations and randomized memberships to examine the contributions of the decoder and biological pathway structure. The main prediction table covers all four tissues; detailed control tables are shown for Kidney, with the remaining tissues in Appendix D. Pathway score and spatial analyses on Kidney further examine how query corrections affect intermediate predictions.

4.1 Experimental setup

Data and preprocessing. We evaluate HyBoxST on Kidney, Colorectum, Skin, and Lung from HEST-1K (Jaume et al., 2024a), containing 23, 14, 46, and 16 slides, respectively. Spot images are cropped using the slide resolution and spot diameter with a minimum physical field of view; paired niche images cover the center spot and six surrounding neighbors. Both inputs are resized to 224×224 pixels. Each tissue uses a fixed panel of 200 genes selected for high mean expression and variability, with spot targets transformed as $\log_2(1 + x)$. Panel selection uses expression statistics from all slides, including test slides; the selection procedure and target construction are detailed in Appendix C.1.

Pathway resources. We intersect Reactome and Hallmark gene sets with each tissue's prediction panel and retain pathways containing at least five selected genes. The resulting memberships define the pathway boxes' supervision, the pathway expression targets, and the decoder's mapping from pathways to genes. Reference pathway scores average the transformed expression of their selected member genes. Appendix C.1 reports pathway counts and gene coverage for each tissue.

Evaluation protocol. Each tissue uses five predefined slide holdouts, allocating approximately 80%, 10%, and 10% of slides to training, validation, and testing within each run (Zhang et al., 2026). The three sets are disjoint within a run, and a separate model is trained for each partition. We report means and standard deviations across the five runs. PCC@200 measures the mean per gene Pearson correlation across test spots over the full prediction panel; MSE and MAE evaluate expression error on the same panel. Supplementary PCC@10 and PCC@50 average the highest 10 and 50 gene correlations for each model and run, so their gene subsets can differ between methods. Higher PCC and lower errors indicate better predictions.

Comparison methods. We compare HyBoxST with TRIPLEX, StNet, BLEEP, Stem, and HyperST. Their results in Table 1 are quoted from Zhang et al. (2026). These published results provide a benchmark comparison; our component ablations and randomized membership controls examine the proposed decoder under shared experimental settings.

Table 1: Prediction on four tissue datasets, with PCC@200 as the primary correlation metric. Baseline means and standard deviations are quoted from Zhang et al. (2026); HyBoxST values summarize our five runs. PCC@10/50 use model-specific subsets ranked by test correlation. Bold and underline indicate the best and second-best means within each tissue, including ties at the displayed precision.

Dataset	Model	PCC@200 ↑	PCC@50 ↑	PCC@10 ↑	MSE ↓	MAE ↓
Kidney	TRIPLEX	0.351 ± 0.066	0.485 ± 0.084	0.579 ± 0.095	1.122 ± 0.204	0.855 ± 0.104
	StNet	0.305 ± 0.064	0.435 ± 0.095	0.523 ± 0.105	1.167 ± 0.217	0.847 ± 0.078
	BLEEP	0.310 ± 0.071	0.434 ± 0.102	0.518 ± 0.112	1.233 ± 0.244	0.865 ± 0.085
	Stem	0.271 ± 0.059	0.414 ± 0.084	0.535 ± 0.111	1.380 ± 0.347	0.911 ± 0.115
	HyperST	$\underline{0.390 \pm 0.070}$	$\underline{0.526 \pm 0.088}$	$\underline{0.617 \pm 0.094}$	$\underline{1.077 \pm 0.155}$	$\mathbf{0.817 \pm 0.058}$
	HyBoxST	$\mathbf{0.428 \pm 0.048}$	$\mathbf{0.618 \pm 0.054}$	$\mathbf{0.747 \pm 0.073}$	$\mathbf{1.020 \pm 0.122}$	$\underline{0.824 \pm 0.047}$
Colorectum	TRIPLEX	$\underline{0.462 \pm 0.191}$	0.624 ± 0.154	0.701 ± 0.128	1.869 ± 0.803	1.056 ± 0.239
	StNet	0.419 ± 0.176	0.570 ± 0.142	0.646 ± 0.134	1.686 ± 0.373	1.023 ± 0.134
	BLEEP	0.382 ± 0.160	0.556 ± 0.120	0.637 ± 0.112	2.038 ± 0.587	1.096 ± 0.164
	Stem	0.399 ± 0.166	0.573 ± 0.130	0.670 ± 0.116	1.788 ± 0.418	1.032 ± 0.138
	HyperST	$\mathbf{0.477 \pm 0.184}$	$\underline{0.642 \pm 0.128}$	$\underline{0.721 \pm 0.105}$	$\underline{1.498 \pm 0.456}$	$\underline{0.958 \pm 0.158}$
	HyBoxST	0.438 ± 0.170	$\mathbf{0.646 \pm 0.136}$	$\mathbf{0.734 \pm 0.104}$	$\mathbf{1.481 \pm 0.474}$	$\mathbf{0.941 \pm 0.164}$
Skin	TRIPLEX	0.740 ± 0.142	0.799 ± 0.114	0.831 ± 0.094	0.981 ± 0.466	0.685 ± 0.205
	StNet	0.726 ± 0.140	0.779 ± 0.117	0.804 ± 0.105	0.993 ± 0.469	0.689 ± 0.198
	BLEEP	0.704 ± 0.145	0.761 ± 0.123	0.788 ± 0.111	1.117 ± 0.540	0.701 ± 0.221
	Stem	0.687 ± 0.138	0.748 ± 0.113	0.782 ± 0.094	1.276 ± 0.703	0.730 ± 0.261
	HyperST	$\underline{0.758 \pm 0.129}$	$\underline{0.812 \pm 0.102}$	$\underline{0.839 \pm 0.086}$	$\underline{0.932 \pm 0.418}$	$\mathbf{0.657 \pm 0.182}$
	HyBoxST	$\mathbf{0.803 \pm 0.072}$	$\mathbf{0.857 \pm 0.060}$	$\mathbf{0.876 \pm 0.052}$	$\mathbf{0.927 \pm 0.130}$	$\underline{0.661 \pm 0.078}$
Lung	TRIPLEX	0.393 ± 0.272	0.499 ± 0.272	0.567 ± 0.247	1.537 ± 1.307	0.849 ± 0.446
	StNet	0.355 ± 0.253	0.464 ± 0.267	0.526 ± 0.247	1.660 ± 1.258	0.899 ± 0.422
	BLEEP	0.311 ± 0.230	0.412 ± 0.235	0.488 ± 0.212	1.803 ± 1.507	0.891 ± 0.465
	Stem	0.351 ± 0.241	0.469 ± 0.250	0.546 ± 0.224	1.709 ± 1.365	0.866 ± 0.485
	HyperST	$\underline{0.459 \pm 0.282}$	$\underline{0.568 \pm 0.260}$	$\underline{0.637 \pm 0.225}$	$\underline{1.182 \pm 0.873}$	$\underline{0.757 \pm 0.352}$
	HyBoxST	$\mathbf{0.592 \pm 0.160}$	$\mathbf{0.752 \pm 0.131}$	$\mathbf{0.819 \pm 0.083}$	$\mathbf{0.811 \pm 0.409}$	$\mathbf{0.602 \pm 0.186}$

Implementation. We use UNI with LoRA applied to attention QKV projections in the final 11 encoder blocks. All four tissues share the same training hyperparameters: AdamW with learning rate 10^{-4}, batch size 128, and up to 100 epochs, with early stopping after 10 epochs without improvement in validation loss. Within each tissue, the full model, ablations, and randomized controls use the same partitions and gene targets. Evaluation uses one original view per image input without test time augmentation. Decoder parameters, loss weights, checkpoint selection, and prediction postprocessing are specified in Appendix C.2.

4.2 Main prediction results

Table 1 compares gene expression prediction across four tissues, with relative changes calculated from the reported means against the strongest baseline for each metric. HyBoxST improves PCC@200 by approximately 9.7%, 5.9%, and 29.0% on Kidney, Skin, and Lung, respectively. Lung shows the largest improvement, with MSE and MAE also reduced by 31.4% and 20.5%. MSE decreases by 5.3% and 0.5% on Kidney and Skin, respectively, while Colorectum achieves reductions of 1.1% in MSE and 1.8% in MAE. Although PCC@200 decreases by 8.2% on Colorectum and MAE increases slightly on Kidney and Skin, the correlation gains across the other three tissues, particularly Lung, demonstrate the method's benefits across multiple tissue datasets.

Spatial expression maps for UMOD and PODXL are provided in Appendix D.6, Figure 4.

4.3 Component ablations and pathway prior controls

We examine whether query correction improves prediction, how box queries and membership supervision contribute, and what the auxiliary objectives provide. Table 2 reports Kidney results; control definitions and the other tissues are given in Appendices C.3, C.4, C.5, and D.

Table 2: Component ablations on Kidney. Values are means with standard deviations. Bold and underline indicate the best and second-best means in each column, including ties at the displayed precision.

Variant	PCC@200 ↑	PCC@50 ↑	PCC@10 ↑	MSE ↓	MAE ↓
HyBoxST	$\mathbf{0.428 \pm 0.048}$	$\mathbf{0.618 \pm 0.054}$	$\mathbf{0.747 \pm 0.073}$	$\mathbf{1.020 \pm 0.122}$	$\mathbf{0.824 \pm 0.047}$
w/o query box	0.402 ± 0.059	0.578 ± 0.062	0.707 ± 0.081	1.139 ± 0.125	0.839 ± 0.052
MLP query	0.403 ± 0.049	0.581 ± 0.054	0.711 ± 0.069	1.127 ± 0.128	0.834 ± 0.051
w/o box loss	0.416 ± 0.056	0.589 ± 0.059	0.722 ± 0.086	1.121 ± 0.128	0.833 ± 0.051
w/o pathway loss	0.420 ± 0.057	0.600 ± 0.057	0.728 ± 0.083	1.116 ± 0.126	0.831 ± 0.053
w/o direct pathway loss	$\underline{0.424 \pm 0.051}$	0.613 ± 0.056	0.739 ± 0.071	$\underline{1.108 \pm 0.129}$	$\underline{0.829 \pm 0.048}$
w/o decoder calibration	$\mathbf{0.428 \pm 0.053}$	$\underline{0.617 \pm 0.057}$	$\underline{0.745 \pm 0.074}$	1.123 ± 0.126	0.837 ± 0.050

Contribution of query correction. To assess the contribution of image conditioned pathway correction, we remove the query branch and retrain the model. This reduces Kidney PCC@200 from 0.428 to 0.402 and increases MSE from 1.020 to 1.139. All five metrics worsen across the four tissues, supporting the contribution of query correction within the structured decoder.

Box queries and membership supervision. To separate the effect of box parameterization from membership supervision, we compare an MLP query with approximately matched prediction parameter count, box queries without membership loss, and the full model. Their Kidney PCC@200 values are 0.403, 0.416, and 0.428. The first comparison supports the box query parameterization relative to the tested MLP; the second shows the additional benefit of membership supervision within the box model. The same ordering holds for mean PCC@200 across all four tissues, although not every intermediate comparison improves every metric (Appendix D.4).

Auxiliary objectives and calibration. We next test whether supervision of intermediate pathway scores and output calibration contribute to gene prediction. Removing corrected or direct pathway supervision lowers Kidney PCC@200 to 0.420 or 0.424. Removing decoder calibration leaves PCC@200 unchanged at the displayed precision (0.428) but increases MSE to 1.123. These results support supervising intermediate pathway scores during training and calibrating the output expression scale. The comparisons describe observed means and do not establish statistical significance.

Pathway prior controls. To test whether prediction benefits from biological gene identities beyond pathway sizes and membership counts, we compare two randomized membership controls. Gene label shuffling permutes mask columns while keeping the prediction gene order fixed, preserving pathway sizes but reassigning memberships. Degree preserving shuffling additionally retains each named gene's membership count. Both use seed 42 (Appendix C.4). Each control uses its own mask for geometric supervision, pathway targets, and gene decoding, while retaining the same gene expression targets. Table 3 reports Kidney results.

Table 3: Pathway mask controls on Kidney. Values are means with standard deviations. Bold and underline indicate the best and second-best means in each column, including ties at the displayed precision.

Mask setting	PCC@200 ↑	PCC@50 ↑	PCC@10 ↑	MSE ↓	MAE ↓
Real pathway mask	$\mathbf{0.428 \pm 0.048}$	$\mathbf{0.618 \pm 0.054}$	$\mathbf{0.747 \pm 0.073}$	$\mathbf{1.020 \pm 0.122}$	$\mathbf{0.824 \pm 0.047}$
Shuffled mask	0.409 ± 0.051	0.585 ± 0.059	0.714 ± 0.071	1.137 ± 0.129	0.834 ± 0.053
Degree preserving shuffle	$\underline{0.414 \pm 0.048}$	$\underline{0.610 \pm 0.057}$	$\underline{0.719 \pm 0.071}$	$\underline{1.129 \pm 0.127}$	$\underline{0.831 \pm 0.051}$

The real mask has better means on all five metrics: Kidney PCC@200 is 0.428, versus 0.409 and 0.414 for the two controls. This direction holds on all four tissues (Appendix D), supporting the additional predictive value of biological memberships beyond the tested connectivity statistics. Each shuffle uses one fixed mask per tissue, so these runs do not measure variability across random masks. We next examine how the model's pathway scores correspond to reference scores defined by real memberships.

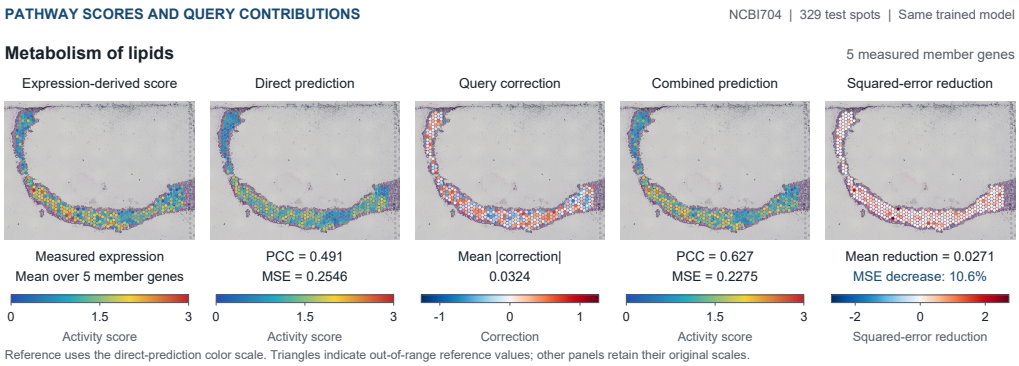

Figure 2: Query correction for Metabolism of lipids on NCBI704 (329 spots; five measured member genes). Columns: reference, direct prediction, correction, combined prediction, and squared error reduction. Blue/red denote decreases/increases in correction and worsening/improvement in error reduction. Reference and prediction maps share the 0–3 scale; correction and error reduction use separate signed scales. *Activity score* means member expression average (Equation 8); triangles mark out-of-range reference values.

4.4 Pathway score and spatial analysis

Pathway score alignment. We next examine whether the pathway scores that guide gene decoding recover observed member expression. On Kidney, internal pathway scores have pooled PCC values of 0.673 with real memberships, 0.274 with gene label shuffling, and 0.478 with degree preserving shuffling; the corresponding MAEs are 0.580, 0.804, and 0.724. Scores aggregated from final gene predictions follow the same direction (Appendix D.5). This agreement supports correspondence to the measured pathway expression targets. The metrics pool spots and pathways across five runs, so they do not isolate spatial accuracy within individual pathways. Shuffled models also learn different pathway targets; this comparison assesses agreement with real pathway references, while the gene prediction controls evaluate a shared target.

Spatial contribution of query corrections. The retrained ablations evaluate performance after removing query correction; here we hold one trained model fixed to examine how its correction changes predictions across tissue locations. Figure 2 decomposes the Metabolism of lipids score on NCBI704 into direct prediction and query correction. Relative to measured member expression, adding the correction raises PCC from 0.491 to 0.627 and lowers MSE from 0.2546 to 0.2275 (10.6%). The error reduction map shows both improved and worsened spots, illustrating a spatially varying adjustment with a net reduction in this example's error. The analysis concerns measured member expression and does not establish whole pathway activity. Evaluation details and two additional examples appear in Appendix D.7.

5 Conclusion

HyBoxST incorporates gene to pathway memberships into expression prediction through a structured decoder. Image context generates query corrections to pathway scores, which are propagated to member genes and combined with a direct branch that allows expression beyond the selected pathways to be modeled. Across four tissues, internal controls support improvements from query correction and the additional predictive value of biological memberships beyond the tested mask statistics. Comparisons with published baseline results show gains whose extent varies by tissue and metric. Pathway analyses further connect the corrected scores to observed member expression and show how query corrections change predictions across tissue locations.

AI USE STATEMENT

Generative AI tools were used for language polishing and to check the consistency of the manuscript's organization and structure. The authors reviewed and verified all content and take full responsibility for the final manuscript.

ETHICS STATEMENT

This work uses publicly available or appropriately licensed histology and spatial transcriptomics datasets. The proposed method is intended for research use in computational pathology and should not be interpreted as a standalone clinical diagnostic system.

REPRODUCIBILITY STATEMENT

Appendix A specifies the decoder and objectives. Appendix C.2 provides the training configuration, initialization, checkpoint rule, single-view inference, and postprocessing. Appendix C describes the partitions, panel selection, and pathway resources; Appendix D details the controls. Baseline table entries are quoted from Zhang et al. (2026). Code and reproduction instructions are planned for release on GitHub.

REFERENCES

Richard J. Chen, Tong Ding, Ming Y. Lu, Drew F. K. Williamson, Guillaume Jaume, Andrew H. Song, et al. Towards a general-purpose foundation model for computational pathology. *Nature Medicine*, 30(3):850–862, 2024. doi: 10.1038/s41591-024-02857-3.

Joan Choo, Joseph J. Zhao, Mai Chan Lau, Alessandra Raimondi, Willa Wen-You Yim, Sruthi Ranganathan, Kexin Zhu, Crystal Tze Ying Tan, Hui Xian Chin, Chong Boon Teo, Khi Yung Fong, Ryan Yong Kiat Tay, Jia-Ying Joey Lee, Lit-Hsin Loo, Paolo Manca, Federica Morano, Michele Prisciandaro, Giovanni Randon, Camilla Damonte, Elisa Micarelli, Guido Leoni, Elisa Scarselli, Sara Lonardi, Chiara Cremolini, Federica Marmorino, Jeffrey Chun Tatt Lim, Zhen Wei Neo, Felicia Wee, Li Yen Chong, Craig Ryan Joseph, Akhila Balachander, Diana GS Lim, You Yi Hwang, Laurent Renia, Subhra Kumar Biswas, David Tan, Joe P. S. Yeong, Raghav Sundar, and Filippo Pietrantonio. Spatial predictors of response to chemo-immunotherapy in microsatellite stable metastatic colorectal cancer. *Nature Communications*, 17(1):6294, April 2026. ISSN 2041-1723. doi: 10.1038/s41467-026-72204-2. URL http://dx.doi.org/10.1038/s41467-026-72204-2.

Youngmin Chung, Ji Hun Ha, Kyeong Chan Im, and Joo Sang Lee. Accurate spatial gene expression prediction by integrating multi-resolution features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11591–11600, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Chung_Accurate_Spatial_Gene_Expression_Prediction_by_Integrating_Multi-Resolution_Features_CVPR_2024_paper.html.

Shib Sankar Dasgupta, Michael Boratko, Dongxu Zhang, Luke Vilnis, Xiang Lorraine Li, and Andrew McCallum. Improving local identifiability in probabilistic box embeddings. In *Advances in Neural Information Processing Systems*, volume 33, pp. 182–192, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/01c9d2c5b3ff5cbba349ec39a570b5e3-Abstract.html.

Octavian Ganea, Gary Becigneul, and Thomas Hofmann. Hyperbolic entailment cones for learning hierarchical embeddings. In *International Conference on Machine Learning*, pp. 1646–1655, 2018. URL https://proceedings.mlr.press/v80/ganea18a.html.

Marc Gillespie, Bijay Jassal, Ralf Stephan, Marija Milacic, Karen Rothfels, Andrea Senff-Ribeiro, Johannes Griss, Cristoffer Sevilla, Lisa Matthews, Chuqiao Gong, Chuan Deng, Thawfeek Varusai, Eliot Ragueneau, Yusra Haider, Bruce May, Veronica Shamovsky, Joel Weiser, Timothy Brunson, Nasim Sanati, Liam Beckman, Xiang Shao, Antonio Fabregat, Konstantinos Sidiropoulos,

Julieth Murillo, Guilherme Viteri, Justin Cook, Solomon Shorser, Gary Bader, Emek Demir, Chris Sander, Robin Haw, Guanming Wu, Lincoln Stein, Henning Hermjakob, and Peter D'Eustachio. The reactome pathway knowledgebase 2022. *Nucleic Acids Research*, 50(D1):D687–D692, 2022. doi: 10.1093/nar/gkab1028.

Bryan He, Ludvig Bergenstrahle, Linnea Stenbeck, Abubakar Abid, Alma Andersson, Åke Borg, Jonas Maaskola, Joakim Lundeberg, and James Zou. Integrating spatial gene expression and breast tumour morphology via deep learning. *Nature Biomedical Engineering*, 4(8):827–834, 2020. doi: 10.1038/s41551-020-0578-x.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Susu Hu, Qinghe Zeng, Nithya Bhasker, Jakob Nikolas Kather, and Stefanie Speidel. Histoprism: Unlocking functional pathway analysis from pan-cancer histology via gene expression prediction. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2601.21560.

Tinglin Huang, Tianyu Liu, Mehrtash Babadi, Wengong Jin, and Zhitao Ying. Scalable generation of spatial transcriptomics from histology images via whole-slide flow matching. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 25550–25565. PMLR, 2025. URL https://proceedings.mlr.press/v267/huang25t.html.

Guillaume Jaume, Paul Doucet, Andrew H. Song, Ming Y. Lu, Cristina Almagro-Perez, Sophia J. Wagner, Anurag J. Vaidya, Richard J. Chen, Drew F. K. Williamson, Ahrong Kim, and Faisal Mahmood. Hest-1k: A dataset for spatial transcriptomics and histology image analysis. In *Advances in Neural Information Processing Systems*, volume 37, pp. 53798–53833, 2024a. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/60a899cc31f763be0bde781a75e04458-Abstract-Datasets_and_Benchmarks_Track.html.

Guillaume Jaume, Anurag Vaidya, Richard J. Chen, Drew F. K. Williamson, Paul Pu Liang, and Faisal Mahmood. Modeling dense multimodal interactions between biological pathways and histology for survival prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11579–11590, 2024b. doi: 10.1109/CVPR52733.2024.01100.

María José Jiménez-Santos, Santiago García-Martín, Marcos Rubio-Fernández, Gonzalo Gómez-López, and Fátima Al-Shahrour. Spatial transcriptomics in breast cancer reveals tumour microenvironment-driven drug responses and clonal therapeutic heterogeneity. *NAR Cancer*, 6 (4):zcae046, 2024. ISSN 2632-8674. doi: 10.1093/narcan/zcae046. URL http://dx.doi.org/10.1093/narcan/zcae046.

Arthur Liberzon, Chet Birger, Helga Thorvaldsdottir, Mahmoud Ghandi, Jill P. Mesirov, and Pablo Tamayo. The molecular signatures database hallmark gene set collection. *Cell Systems*, 1(6): 417–425, 2015. doi: 10.1016/j.cels.2015.12.004.

Sejuti Majumder, Saarthak Kapse, Moinak Bhattacharya, Xuan Xu, Alisa Yurovsky, and Prateek Prasanna. PEaRL: Pathway-enhanced representation learning for gene and pathway expression prediction from histology. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 8052–8062, March 2026.

Takashi Monjo, Masaru Koido, Shinichi Nagasawa, Yutaka Suzuki, and Yoichiro Kamatani. Efficient prediction of a spatial transcriptomics profile better characterizes breast cancer tissue sections without costly experimentation. *Scientific Reports*, 12(1):4133, 2022. doi: 10.1038/s41598-022-07685-4.

Maximilian Nickel and Douwe Kiela. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *International Conference on Machine Learning*, pp. 3779–3788, 2018. URL https://proceedings.mlr.press/v80/nickel18a.html.

Hongyu Ren, Weihua Hu, and Jure Leskovec. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=BJgr4kSFDS.

Hang Shi, Changxi Chi, Peng Wan, Daoqiang Zhang, and Wei Shao. Multi-modal topology-embedded graph learning for spatially resolved genes prediction from pathology images with prior gene similarity information. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20810–20819, June 2025.

Patrik L. Stahl, Fredrik Salmen, Sanja Vickovic, Anna Lundmark, Jose Fernandez Navarro, Jens Magnusson, Stefania Giacomello, Michaela Asp, Jakub O. Westholm, Mikael Huss, Annelie Mollbrink, Sten Linnarsson, Simone Codeluppi, Åke Borg, Fredrik Ponten, Paul Igor Costea, Pelin Sahlen, Jan Mulder, Olaf Bergmann, Joakim Lundeberg, and Jonas Frisen. Visualization and analysis of gene expression in tissue sections by spatial transcriptomics. *Science*, 353(6294): 78–82, 2016. doi: 10.1126/science.aaf2403.

Luke Vilnis, Xiang Li, Shikhar Murty, and Andrew McCallum. Probabilistic embedding of knowledge graphs with box lattice measures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pp. 263–272, 2018. doi: 10.18653/v1/P18-1025.

Cameron G. Williams, Hyun Jae Lee, Takahiro Asatsuma, Roser Vento-Tormo, and Ashraful Haque. An introduction to spatial transcriptomics for biomedical research. *Genome Medicine*, 14(1):68, 2022. doi: 10.1186/s13073-022-01075-1.

Ronald Xie, Kuan Pang, Sai Chung, Catia Perciani, Sonya MacParland, Bo Wang, and Gary Bader. Spatially resolved gene expression prediction from histology images via bi-modal contrastive learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 70626–70637, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/df656d6ed77b565e8dcdfbf568aead0a-Abstract.html.

Zhiwen Xu, Xiaoming Yan, Chengkun Wu, Juan Chen, Haoang Chi, and Liyang Xu. Gene ontology-guided hierarchical spatial gene expression prediction from histopathology images. *arXiv preprint arXiv:2608.00405*, 2026. URL https://arxiv.org/abs/2608.00405.

Chen Zhang, Yilu An, Ying Chen, Hao Li, Xitong Ling, Lihao Liu, Junjun He, Yuxiang Lin, Zihui Wang, and Rongshan Yu. Hyperst: Hierarchical hyperbolic learning for spatial transcriptomics prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5730–5739, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Zhang_HyperST_Hierarchical_Hyperbolic_Learning_for_Spatial_Transcriptomics_Prediction_CVPR_2026_paper.html.

L. Zhang, D. Chen, D. Song, X. Liu, Y. Zhang, X. Xu, and X. Wang. Clinical and translational values of spatial transcriptomics. *Signal Transduction and Targeted Therapy*, 7(1):111, 2022. doi: 10.1038/s41392-022-00960-w.

Sichen Zhu, Yuchen Zhu, Molei Tao, and Peng Qiu. Diffusion generative modeling for spatially resolved gene expression inference from histology images. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2501.15598.

A Additional Method Details

A.1 Spatial context representation

Training expression helps the images learn molecularly informative features through the inherited alignment objective $\mathcal{L}_{\text{align}}$ (Zhang et al., 2026). Spot expression Y_i and niche expression Y_i^n are encoded and aligned with the corresponding image features in a Lorentz space using contrastive and entailment terms. Niche expression is formed by averaging original counts over the retained neighborhood, including the center spot, and then applying $\log_2(1+x)$. It is used only for alignment. These training terms update the visual transformations in Equation 1; expression features are not concatenated into h_i. The decoder receives the image features before the Lorentz mapping, and query matching uses the separate Euclidean box space.

The spatial module uses spot and niche images to capture local morphology and its surroundings. A pathology encoder with LoRA adaptation extracts the two visual representations (Hu et al., 2022; Chen et al., 2024):

$$I_i^s, I_i^n = E_{\text{img}}(X_i^s, X_i^n). \tag{10}$$

The corresponding spot and niche gene embeddings are obtained from the observed expression profiles during training:

$$G_i^s, G_i^n = E_{\text{gene}}(Y_i, Y_i^n). \tag{11}$$

These G_i^s, G_i^n encode sample expression for alignment; they are distinct from the learned gene points e_g used to supervise pathway boxes. Following the inherited spatial alignment module (Zhang et al., 2026), the features are projected to a Lorentz space with learnable curvature (Nickel & Kiela, 2018; Ganea et al., 2018). The alignment module combines contrastive matching between image and gene representations with entailment regularization, encouraging multi level visual features to encode molecularly informative spatial context. Let $\mathcal{L}_{\text{align}}$ denote the resulting hierarchical alignment loss. The alignment loss is computed on the Lorentz representations, but the decoder receives the Euclidean features before the exponential map (with learned scaling when enabled). These image features are concatenated as

$$h_i = [\tilde{I}_i^s; \tilde{I}_i^n], \tag{12}$$

where $\tilde{I}_i^s$ and $\tilde{I}_i^n$ are the spot and niche image features before the Lorentz mapping, concatenated into h_i to supply the query, direct pathway, and direct gene heads in Section 3.

The alignment implementation uses four contrastive directions: spot image to spot expression, spot expression to spot image, niche image to spot expression, and niche expression to spot image. In each direction, the paired sample is the positive and the other samples in the batch provide negatives. With $\ell_{ij} = -t\, d_{\mathcal{L}}(z_i, z_j)$, each term is the cross entropy $-\log[\exp(\ell_{ii})/\sum_j \exp(\ell_{ij})]$, averaged over queries. The four terms are averaged with factor $1/4$. The learned inverse temperature t starts at $1/0.07$ and its log is capped at 4.6052; curvature starts at one and is constrained to $[0.1, 10]$.

For a directed pair (a, b), the entailment penalty is $[\text{angle}(a, b) - k\,\text{aperture}(a)]_+$. It uses $k = 0.7$ for spot image to spot expression and niche image to niche expression, and $k = 1.2$ for spot image to niche image and spot expression to niche expression. Each term is averaged over the batch; their sum is multiplied by $1/2$. The resulting entailment loss is weighted by 0.4 and added to the contrastive loss to form $\mathcal{L}_{\text{align}}$, which receives the outer weight 0.2 in the full objective. The decoder uses the projected image features before the learned scaling and exponential map in this configuration.

A.2 Membership supervision and query geometry

The gene points and pathway boxes are trained against the fixed annotations M. For a gene point e_g, the point to box violation distance is

$$d_{pg} = \| \max(l_p^B - e_g, 0) + \max(e_g - u_p^B, 0) \|_2. \tag{13}$$

The distance d_{pg} is zero for a contained gene point and positive outside the box; it supplies the membership penalty below. Let $\Omega^+ = \{(p, g) : M_{pg} = 1\}$ and $\Omega^- = \{(p, g) : M_{pg} = 0\}$. The membership objective is

$$\mathcal{L}_{\text{box}} = \frac{1}{|\Omega^+|} \sum_{(p,g)\in\Omega^+} d_{pg} + \frac{\lambda_{\text{neg}}}{|\Omega^-|} \sum_{(p,g)\in\Omega^-} \max(0, m - d_{pg}). \tag{14}$$

The first term penalizes annotated members outside their boxes; the second penalizes nonmembers whose distance is below the margin $m > 0$. The resulting $\mathcal{L}_{\text{box}}$ enters Equation 9 to train the gene points and pathway bounds. Gene points participate only in this objective. Pathway boxes also receive gradients through query matching and expression prediction, so the learned regions are constrained by both the annotations and the prediction task. Exact containment or exclusion is not guaranteed; Appendix B.1 describes the flat region for nonmembers inside a box.

The pathway boxes are shared learned parameters, while queries depend on individual images. A gene point contained in two pathway boxes establishes a nonempty intersection between those boxes. The converse does not hold: two boxes may intersect in a region containing no gene point. The representation therefore supports shared membership without making geometric intersection equivalent to an observed shared gene. Appendix A.3 expands the smoothed query matching calculation in Equation 4.

A.3 QUERY OVERLAP COMPUTATION

For one spot and pathway, the stabilized widths used in Equation 4 are

$$t_{ipd} = \min(u^Q_{i,d}, u^B_{p,d}) - \max(l^Q_{i,d}, l^B_{p,d}), \tag{15}$$

$$\tilde{v}_{ipd} = \tau \log(1 + \exp(t_{ipd}/\tau)) + \epsilon, \tag{16}$$

$$\tilde{q}_{id} = \max(u^Q_{i,d} - l^Q_{i,d}, \epsilon) + \epsilon, \qquad r_{ipd} = \tilde{v}_{ipd}/\tilde{q}_{id} > 0. \tag{17}$$

The signed width t_{ipd} measures interval overlap in one coordinate; negative values indicate a gap. Smoothing gives the positive width $\tilde{v}_{ipd}$, and division by the query width $\tilde{q}_{id}$ produces the relative match r_{ipd}. Here $\tau > 0$ controls smoothing and $\epsilon > 0$ stabilizes division and logarithms. The lower bound on the query width matches the implementation; it is inactive under the reported minimum half size of 10^{-3} and stabilizer 10^{-6}. Equation 4 combines these ratios across coordinates into o_{ip}. Collecting the scores across pathways gives o_i, which produces the correction ΔP_i in Equation 5. Appendix B.2 analyzes its geometric mean formulation and clipping.

The minimum query half size ϵ_{size} and the overlap stabilizer ϵ are distinct constants: the former keeps query boxes nondegenerate, whereas the latter stabilizes the logarithms and ratios in Equation 4. Pathway half sizes are also parameterized by softplus plus a positive lower bound.

A.4 MEMBERSHIP CONSTRAINED DECODING AND CALIBRATION

The corrected pathway scores are aggregated through the membership constrained weights in Equation 6 and combined with the direct gene prediction by Equation 7. Appendix B.3 derives the weight constraints and propagation of pathway corrections.

The final affine layer is trained jointly with the decoder:

$$\hat{Y}_{ig} = \alpha_g \hat{Y}^{\text{raw}}_{ig} + \beta_g, \qquad \alpha_g = \text{softplus}(\bar{\alpha}_g) > 0. \tag{18}$$

Here $\hat{Y}^{\text{raw}}_{ig}$ is the gated estimate from Equation 7; the calibrated $\hat{Y}_{ig}$ is used in the prediction objective below and passed to the evaluation postprocessing described in Appendix C.2. The gene gates γ_g and calibration parameters α_g, β_g are shared across spots. The positive scale preserves the direction of changes in the raw prediction when parameters are held fixed; Appendix B.6 derives its effect on correlation.

In the decoder calibration ablation, this layer is disabled during training, so its output is the uncalibrated gated prediction. This differs from applying or removing a fixed affine map after training: the separately optimized model can produce different correlations. Here calibration refers to this decoder layer, not to the optional postprocessing calibrators in the training code.

A.5 PREDICTION OBJECTIVE

The gene objective supervises the final output with individual prediction errors and penalties on batch statistics:

$$\begin{aligned}\mathcal{L}_{\text{pred}} = {} & \text{MSE}(\hat{Y}, Y) + \lambda_{\text{mae}}\text{MAE}(\hat{Y}, Y) \\ & + \frac{\lambda_\mu}{G}\|\mu(\hat{Y}) - \mu(Y)\|_2^2 + \frac{\lambda_\sigma}{G}\|\sigma_{\text{batch}}(\hat{Y}) - \sigma_{\text{batch}}(Y)\|_2^2.\end{aligned} \tag{19}$$

Here μ and σ_{batch} are the per gene mean and population standard deviation over a mini batch. For a batch $\mathcal{B}$ of size N, the statistics for gene column j are

$$\mu_j(Y) = \frac{1}{N}\sum_{i\in\mathcal{B}} Y_{ij}, \qquad \sigma_{\text{batch},j}(Y) = \sqrt{\frac{1}{N}\sum_{i\in\mathcal{B}}(Y_{ij} - \mu_j(Y))^2}. \tag{20}$$

The same definitions apply to $\hat{Y}$. These statistics constrain the mean and spread of predicted expression during training. Their penalties enter $\mathcal{L}_{\text{pred}}$, which updates both gene branches through the final calibrated prediction in Equation 9. The factors $1/G$ in Equation 19 match mean reduction over genes. The standard deviation uses the population convention, without Bessel's correction.

Algorithm 1 Training with pathway boxes and image conditioned queries

Require: Training samples $(X_i^s, X_i^n, Y_i, Y_i^n)$; fixed mask M; loss weights
Ensure: Trained parameters Θ
Initialize image, alignment, and decoder parameters; pathway boxes and gene points
$M_{\mathrm{norm},pg} \leftarrow M_{pg} / \sum_{g'} M_{pg'}$
Precompute $A_i \leftarrow Y_i M_{\mathrm{norm}}^{\top}$ for training samples (Equation 8)
for each training mini batch $\mathcal{B}$ **do**
    Encode paired spot and niche images and training expression features
    Compute $\mathcal{L}_{\mathrm{align}}$ from their Lorentz representations
    $h_i \leftarrow [\tilde{I}_i^s; \tilde{I}_i^n]$ for $i \in \mathcal{B}$ using image features before the Lorentz map
    $l_p \leftarrow c_p - r_p$, $u_p \leftarrow c_p + r_p$ for all pathways, with positive r_p
    $U_{pg} \leftarrow M_{pg}\,\mathrm{softplus}(\omega_{pg})$
    $W_{pg} \leftarrow U_{pg} / \max(1, \sum_{p'} U_{p'g})$; $\gamma_g \leftarrow \mathbf{1}[\sum_p M_{pg} > 0]\sigma(\theta_g)$
    for each spot $i \in \mathcal{B}$ (computed in parallel) **do**
        $(\mu_i, \rho_i) \leftarrow f_{\mathrm{query}}(h_i)$; $\delta_i \leftarrow \mathrm{softplus}(\rho_i) + \epsilon_{\mathrm{size}}$
        $Q_i \leftarrow [\mu_i - \delta_i, \mu_i + \delta_i]$
        Compute o_{ip} for every pathway by Equation 4
        $P_i^{\mathrm{dir}} \leftarrow f_{\mathrm{path}}(h_i)$; $P_i \leftarrow P_i^{\mathrm{dir}} + \sigma(\eta)(s \odot o_i + b)$
        $\hat{Y}_i^{\mathrm{path}} \leftarrow P_i W$; $\hat{Y}_i^{\mathrm{res}} \leftarrow f_{\mathrm{res}}(h_i)$
        $\hat{Y}_i^{\mathrm{raw}} \leftarrow \gamma \odot \hat{Y}_i^{\mathrm{path}} + (1-\gamma) \odot \hat{Y}_i^{\mathrm{res}}$
        $\hat{Y}_i \leftarrow \mathrm{softplus}(\bar{\alpha}) \odot \hat{Y}_i^{\mathrm{raw}} + \beta$
    end for
    Compute $\mathcal{L}_{\mathrm{pred}}$ on the batch gene predictions
    $\mathcal{L}_{\mathrm{path}} \leftarrow \mathrm{MSE}(P_{\mathcal{B}}, A_{\mathcal{B}})$;
    $\mathcal{L}_{\mathrm{dir}} \leftarrow \mathrm{MSE}(P_{\mathcal{B}}^{\mathrm{dir}}, A_{\mathcal{B}})$
    Compute $\mathcal{L}_{\mathrm{box}}$ once for the batch using Equation 14
    Form $\mathcal{L}$ using Equation 9
    Update Θ by backpropagating $\mathcal{L}$
end for

A.6 Training and prediction procedure

Algorithm 1 summarizes training with a fixed membership matrix. The same selected matrix is used to construct pathway targets, supervise gene containment, and constrain gene decoding. For randomized controls, it is replaced consistently at all three locations. The direct gene branch is optimized through the final gene loss, without a separately constructed residual target.

At prediction time, freeze the parameters and use only the image feature and decoder computations. Expression encoders, target construction, alignment loss, and box membership loss are unnecessary for producing predictions. The learned pathway bounds, masked weights, and gene gates can be cached. This describes prediction without auxiliary loss evaluation; the current evaluation forward pass also computes losses for reporting, which adds the training-side computations discussed below.

B Mathematical Properties and Computational Complexity

This appendix derives properties of the pathway scores and decoder defined in Section 3. The results explain how averaging, geometric matching, and query corrections affect predictions. The derivations are algebraic consequences of the model; the final subsection analyzes computational complexity.

B.1 Local geometry of the membership loss

The point to box distance is zero throughout the box interior. For a nonmember point strictly inside its box, sufficiently small perturbations of the point or box bounds preserve containment, so its hinge term remains m and has zero local gradient with respect to both. For example, a box $[-1, 1]$ and a nonmember point $e = 0$ give a constant negative term m for every $e \in (-1, 1)$. This is a flat region of the stated objective, not evidence that the full model cannot learn: shared gene points and pathway boxes also receive updates from other pairs and objectives. The reported experiments

do not quantify the frequency or persistence of such interior negatives, so the loss should not be interpreted as guaranteeing their exclusion.

B.2 Geometric mean overlap and numerical stability

Using the stabilized widths in Appendix A.3, write $r_{ipd} = \tilde{v}_{ipd}/\tilde{q}_{id} > 0$.

With hard intersections, the product of normalized widths gives the fraction of a query box covered by its intersection with a pathway box. Smoothing replaces zero intersection widths by positive values. HyBoxST uses the geometric mean of these ratios before clipping, rather than their product:

$$z_{ip} = \frac{1}{D}\sum_d \log r_{ipd}, \qquad \exp(z_{ip}) = \left(\prod_d r_{ipd}\right)^{1/D}. \tag{21}$$

If every dimension has ratio r, the product is r^D whereas the geometric mean is r. For example, $r = 0.5$ and $D = 128$ give a product of approximately 2.94×10^{-39} but a geometric mean of 0.5. The mean therefore avoids exponential decay caused solely by repeating the same overlap ratio across dimensions. Computing it through summed logarithms also avoids forming the small product explicitly.

The intermediate z_{ip} is the log matching score, converted to o_{ip} by clipping and exponentiation below. Because smoothing can make a ratio exceed one, Equation 4 clips the mean log ratio:

$$o_{ip} = \exp(\mathrm{clip}(z_{ip}, -30, 0)), \qquad \exp(-30) \le o_{ip} \le 1. \tag{22}$$

Inside the unclipped region, a partial derivative with respect to an individual ratio is

$$\frac{\partial o_{ip}}{\partial r_{ipd}} = \frac{o_{ip}}{D r_{ipd}}. \tag{23}$$

The dependence is monotone in each ratio in this region. Outside it, clipping saturates the score and its derivative with respect to z_{ip} is zero away from the boundaries. The denominator is the query width, so exchanging the query and pathway boxes generally changes the score. The score is a bounded geometric matching signal, not a symmetric similarity or a calibrated probability of pathway activity.

B.3 Masked weights and propagation to gene predictions

Let $U_{pg} = M_{pg}\mathrm{softplus}(\omega_{pg})$ and $S_g = \sum_p U_{pg}$. Here U_{pg} is the unnormalized contribution allowed by membership, and S_g determines its normalization for gene g. The resulting W_{pg} weights pathway scores in $\hat{Y}_i^{\mathrm{path}} = P_i W$. The weights in Equation 6 satisfy

$$W_{pg} = \frac{U_{pg}}{\max(1, S_g)}, \qquad W_{pg} \ge 0, \qquad \sum_p W_{pg} = \min(S_g, 1) \le 1. \tag{24}$$

The pathway prediction for a gene is therefore a nonnegative combination with total weight at most one. If $S_g \ge 1$, its weights sum to one. If $0 < S_g < 1$, a remaining weight of $1 - S_g$ can be assigned to a zero reference value, yielding a convex combination of the pathway scores and zero. For an uncovered gene, $S_g = 0$ and every $W_{pg} = 0$. Its gate is also zero, so the direct gene branch supplies its prediction before affine calibration.

The gate parameterization in Equation 7 gives $0 \le \gamma_g \le 1$, and the affine scale satisfies $\alpha_g = \mathrm{softplus}(\bar{\alpha}_g) > 0$. Holding the trained parameters and direct gene prediction fixed, consider a change ΔP_{ip} in the intermediate pathway scores. Substitution into the decoder gives the exact change

$$\Delta \widehat{Y}_{ig} = \alpha_g \gamma_g \sum_p W_{pg} \Delta P_{ip}. \tag{25}$$

For the query correction $\Delta P_{ip} = q(s_p o_{ip} + b_p)$, where $q = \sigma(\eta)$, the partial sensitivity to one overlap score is

$$\frac{\partial \widehat{Y}_{ig}}{\partial o_{ip}} = \alpha_g \gamma_g W_{pg} q s_p. \tag{26}$$

Since $\alpha_g > 0$, $\gamma_g \geq 0$, $W_{pg} \geq 0$, and $q = \sigma(\eta) \in (0, 1)$, all factors except s_p are nonnegative. The sign of the query contribution can therefore differ from the sign suggested by overlap alone. A missing membership, $M_{pg} = 0$, makes this direct contribution zero. Applying the triangle inequality and $\sum_p W_{pg} \leq 1$ to Equation 25 gives

$$|\Delta \widehat{Y}_{ig}| \leq \alpha_g \gamma_g \max_p |\Delta P_{ip}|. \tag{27}$$

These statements concern perturbations of intermediate scores within a fixed decoder and describe the model output $\hat{Y}$ before the separate evaluation postprocessing in Appendix C.2. They do not describe the total response to a changed histology image, which can also change the direct pathway and direct gene branches, or the effect of retraining an ablated model.

B.4 When a query correction reduces squared error

Let $e_{ip} = A_{ip} - P_{ip}^{\mathrm{dir}}$ be the error remaining after the direct pathway prediction, and let $\delta_{ip} = q(s_p o_{ip} + b_p)$ be the query correction. Here $q = \sigma(\eta)$ and δ_{ip} is the component ΔP_{ip} from Equation 5, not a query half size. The error after correction is $e_{ip} - \delta_{ip}$. The squared error reduction shown in Figures 2 and 5 is therefore

$$R_{ip} = e_{ip}^2 - (e_{ip} - \delta_{ip})^2 \tag{28}$$

$$= 2e_{ip}\delta_{ip} - \delta_{ip}^2. \tag{29}$$

The quantity R_{ip} is used to visualize and summarize the effect of correction in the reported analysis; it is not fed back into the decoder. For a nonzero correction and nonzero initial error, $R_{ip} > 0$ holds precisely when the correction has the same sign as the initial error and its magnitude is smaller than twice that error:

$$e_{ip}\delta_{ip} > 0, \qquad 0 < |\delta_{ip}| < 2|e_{ip}|. \tag{30}$$

A correction in the wrong direction increases error. A correction in the right direction can also increase error if it overshoots. Averaging over spots produces

$$\mathrm{MSE}_{\mathrm{dir},p} - \mathrm{MSE}_{\mathrm{combined},p} = \frac{1}{N} \sum_i R_{ip}. \tag{31}$$

Consequently, a positive average reduction can coexist with individual spots where prediction becomes worse, as observed in the spatial contribution maps. The mean correction magnitude alone cannot determine whether prediction improves, because improvement also depends on alignment with the initial errors.

B.5 Pathway averaging and prediction error

Let $S_p = \{g : M_{pg} = 1\}$ be the measured members of pathway p, with $n_p = |S_p| > 0$. Define $v_{pg} = M_{pg}/n_p$, so $v_{pg} \geq 0$ and $\sum_g v_{pg} = 1$. Equation 8 can then be written as

$$A_{ip} = \sum_{g=1}^{G} v_{pg} Y_{ig} = \frac{1}{n_p} \sum_{g \in S_p} Y_{ig}. \tag{32}$$

The fixed weights v_{pg} construct the training target A_{ip} by averaging genes within a pathway; they differ from the learned W_{pg} that aggregates pathways into gene predictions. Thus the pathway score is a convex average of member gene expression and lies between the smallest and largest member values at a spot. Since the gene values have already undergone the logarithmic transformation, this operation averages transformed expression; it does not take the logarithm of the average original count.

For pathway scores aggregated from model gene predictions, write $\widehat{A}_{ip} = \sum_g v_{pg} \widehat{Y}_{ig}$ and $e_{ig} = \widehat{Y}_{ig} - Y_{ig}$. The same argument applies to postprocessed predictions by replacing $\widehat{Y}$ with $\hat{Y}^{\mathrm{eval}}$ throughout. Their error satisfies

$$\widehat{A}_{ip} - A_{ip} = \sum_g v_{pg} e_{ig}, \tag{33}$$

$$(\widehat{A}_{ip} - A_{ip})^2 \leq \sum_g v_{pg} e_{ig}^2. \tag{34}$$

The inequality follows from convexity of the square function. Averaging over N spots and P pathways gives

$$\frac{1}{NP}\sum_{i,p}(\widehat{A}_{ip}-A_{ip})^2 \le \frac{1}{N}\sum_{i,g}\kappa_g e_{ig}^2, \qquad \kappa_g=\frac{1}{P}\sum_{p} v_{pg}, \quad \sum_g \kappa_g = 1. \tag{35}$$

The coefficients κ_g summarize how membership and pathway size weight each gene in this error bound; they are analytical quantities, not additional decoder parameters. This is a bound by a weighted gene error, with weights determined by pathway membership and size. It does not imply a bound by the unweighted gene MSE. Errors of opposite sign can cancel within a pathway: two equally weighted member errors t and $-t$ give zero pathway score error despite nonzero gene errors. Accurate aggregate scores therefore complement gene evaluation. Moreover, this derivation concerns $\widehat{A}_{ip}$ aggregated from gene predictions, while the internal pathway prediction P_{ip} is a separate model output.

B.6 Affine calibration and correlation

For one gene, let u_i be a prediction before affine calibration and y_i its target across spots. With a fixed positive scale α and offset β, define $v_i = \alpha u_i + \beta$. For the decoder, u_i corresponds to $\hat{Y}_{ig}^{\mathrm{raw}}$ and v_i to $\hat{Y}_{ig}$ for the chosen gene. When both variables have nonzero variance,

$$\mathrm{Cov}(v,y) = \alpha\,\mathrm{Cov}(u,y), \tag{36}$$

$$\mathrm{Std}(v) = \alpha\,\mathrm{Std}(u), \tag{37}$$

$$\mathrm{Corr}(v,y) = \mathrm{Corr}(u,y). \tag{38}$$

Positive affine calibration can change MSE and MAE while preserving Pearson correlation for fixed predictions. In contrast, its squared error is

$$\mathrm{MSE}(v,y) = \frac{1}{N}\sum_i(\alpha u_i+\beta-y_i)^2, \tag{39}$$

which depends on both calibration parameters. This distinction explains why correlation and expression error need separate evaluation. The invariance applies to a fixed positive affine map; retraining the decoder can change the underlying predictions and hence their correlations.

B.7 Computational complexity

Let B be the batch size, P the number of retained pathways, G the number of predicted genes, D the box dimension, and H the hidden width of each decoder MLP. The input h_i has width $2H$. We analyze the dense tensor operations in the implementation, treating the MLP depth and expansion ratio as fixed. Costs of the shared image encoder and spatial alignment module are denoted separately by C_{shared}; the analysis below isolates molecular decoding and its supervision.

Decoder forward pass. Each of the direct pathway, query, and direct gene MLPs costs $O(BH^2)$. Their output heads cost $O(BHP)$, $O(BHD)$, and $O(BHG)$, respectively. Query overlap compares B query boxes with P pathway boxes over D coordinates, costing $O(BPD)$. Forming the masked weights costs $O(PG)$, and the dense product of pathway scores with these weights costs $O(BPG)$. Gate mixing and affine calibration cost $O(BG)$. Thus, excluding the shared feature extractor, a prediction-only forward pass has cost

$$C_{\mathrm{pred}} = O\big(BH^2 + BH(P+G+D) + BPD + BPG\big). \tag{40}$$

Here the $O(PG)$ weight construction is absorbed by $O(BPG)$ for $B \ge 1$; caching it removes the construction cost but does not change the order of the dense matrix product.

Training overhead from boxes. The membership loss broadcasts all G gene points against all P pathway boxes and reduces over D coordinates. Its cost is $O(PGD)$ per batch, not $O(BPGD)$, because gene points and pathway boxes are shared across spots. Both positive and negative pairs

are evaluated, so the current implementation cannot replace PG by the number of nonzero memberships. Gene and pathway supervision add $O(BG+BP)$. Backpropagation through these operations has the same asymptotic order, giving

$$C_{\text{train}} = C_{\text{shared}}^{\text{train}} + O\big(BH^2 + BH(P+G+D) + BPD + BPG + PGD\big). \tag{41}$$

The fixed pathway targets can be precomputed in $O(NPG)$ for N training spots using the dense membership matrix, as in the data loader, rather than recomputed at every optimization step. The current evaluation forward also evaluates the membership loss and therefore retains its $O(PGD)$ cost; omitting this diagnostic loss gives the prediction-only cost in Equation 40.

Comparison and memory. Relative to the same pathway decoder without a query branch but retaining membership supervision, the query adds $O(BH^2+BHD+BPD)$ forward work. Relative to a direct gene MLP with the same input and hidden width, whose decoder cost is $O(BH^2+BHG)$, the full structured decoder additionally performs pathway prediction, pathway-to-gene aggregation, query matching, and training membership supervision. It also uses three MLP branches instead of one; this constant-factor increase is hidden by the $O(BH^2)$ notation. Box geometry therefore adds real computational cost, but it is linear in D for fixed B, P, G, with no enumeration of box corners and no all-pairs pathway intersection term $O(P^2D)$.

The broadcast overlap and membership computations materialize intermediates of sizes $B \times P \times D$ and $P \times G \times D$, respectively. Their training activation memory is $O(BPD + PGD)$, in addition to MLP activations and the $O(PG)$ dense mask and weight arrays. Chunking these pair computations could reduce peak intermediate memory without changing arithmetic complexity, but is not assumed here. These operation counts do not establish measured runtime or a negligible overhead relative to the image encoder; that requires timing and memory measurements under matched settings.

C Additional Experimental Details

C.1 Datasets, gene selection, and pathway coverage

Table 4 summarizes the tissue datasets and pathway coverage. Each partition allocates approximately 80%, 10%, and 10% of slides to training, validation, and testing, with rounding for small datasets. Within each tissue, all five partitions have the same set sizes. The table reports the numbers of slides in each set for one partition. Across the five partitions, Kidney, Colorectum, Skin, and Lung contain 10, 5, 25, and 10 test-slide occurrences but only 9, 4, 17, and 8 distinct test slides, respectively. A slide can therefore be tested in more than one run; within each run, the three sets remain disjoint.

Table 4: Dataset and pathway resource statistics. Train, validation, and test counts are slides per partition. Each tissue predicts 200 genes; coverage is the proportion assigned to at least one retained pathway.

Tissue	Slides	Train	Val.	Test	Pathways	Covered genes	Coverage
Kidney	23	19	2	2	178	182	91.0%
Colorectum	14	12	1	1	184	176	88.0%
Skin	46	36	5	5	233	171	85.5%
Lung	16	12	2	2	221	169	84.5%

Gene selection first retains genes shared across the slides of a tissue and represented in the expression vocabulary. Within each slide, expression is normalized by total counts and transformed with $\log(1+x)$ for identifying 2,000 highly variable genes. Their union forms the candidate set, after excluding genes with MT, mt, RP, or rp prefixes. Candidates are then ranked separately by mean expression and standard deviation computed from the pooled original expression values. We intersect the high mean and high standard deviation candidate pools, rank their intersection by the sum of the two ranks, and retain 200 genes. Ties are resolved by the larger of the two ranks and then by gene name. This preprocessing pools all slides of a tissue before the training/test partitions are applied, including

slides later used for testing. The resulting 200-gene panel is fixed across the five runs; gene selection is not repeated using training slides alone. Results therefore evaluate prediction conditional on this fixed tissue panel. Because pathway selection is restricted to that panel, it also depends indirectly on the pooled expression statistics. The reported experiments do not evaluate a pipeline that constructs both the gene panel and pathway subset using training slides alone.

Each tissue panel contains 200 distinct genes in the same order used by the pathway membership matrix. The panel is defined at the tissue level and shared across all five runs. The normalization used to identify variable genes is separate from the prediction target transformation. Spot targets use $Y_{ig} = \log_2(1 + x_{ig})$, where x_{ig} is the original count from the filtered input matrix, without total-count normalization. Niche targets first average the original counts over the retained neighborhood, including the center spot, and then apply $\log_2(1 + x)$. Pathway targets instead average already transformed spot expression over member genes.

Pathway resources are constructed by intersecting Reactome and Hallmark gene sets with the selected prediction genes and retaining pathways with at least five members. Pathway reference scores average the transformed expression of available member genes. During training, the same randomly selected image transformation is applied to the paired spot and niche images.

C.2 Training configuration and inference

Table 5 gives the full model configuration. The same training and inference hyperparameters are used across all four tissue datasets. Component removals follow Appendix C.3; the MLP control uses the widths in Table 7.

Table 5: Shared training and decoder configuration for the full model across all four datasets. Loss ablations change the corresponding coefficient as specified in Appendix C.3.

Setting	Value
Optimizer; learning rate	AdamW; 10^{-4}, without a learning rate scheduler
Batch size; maximum epochs	128; 100
Early stopping	Validation loss; patience 10 epochs
Training seed; precision	42 for each partition; automatic mixed precision
EMA decay; test checkpoint	0.999; final saved checkpoint with EMA weights
Image encoder; input resolution	UNI; 224×224 pixels for both image inputs
LoRA rank, scale, dropout	8, 16, 0.1
LoRA target	Attention QKV projections in the final 11 encoder blocks
Shared width H; box dimension D	1,024; 128
Decoder MLPs	Two residual blocks per branch; expansion ratio 2; dropout 0.1
$\lambda_{\text{align}}, \lambda_{\text{path}}, \lambda_{\text{dir}}, \lambda_{\text{box}}$	0.2, 0.1, 0.05, 0.01
$\lambda_{\text{mae}}, \lambda_{\mu}, \lambda_{\sigma}$	0.1, 0.05, 0.02
Membership margin m; negative weight	1.0; 0.1
Overlap temperature τ; stabilizer ϵ	0.1; 10^{-6}
Minimum pathway and query half sizes	10^{-3} for both
Initial query mixing coefficient	$\sigma(\eta) = 0.2$ ($\eta = -1.38629436$)

Architecture and initialization. Each decoder branch applies input layer normalization, a $2H \to H$ linear projection, two residual MLP blocks, output layer normalization, and a linear head. Each residual block applies layer normalization followed by an $H \to 2H \to H$ MLP with GELU and dropout. The heads output P, $2D$, and G values for direct pathway scores, query parameters, and direct gene predictions, respectively. The residual encoders use Xavier uniform initialization for linear weights and zero biases; the separate output heads use the default linear-layer initialization. Pathway centers and gene points are initialized from $\mathcal{N}(0, 0.02^2)$; raw pathway half sizes use $\mathcal{N}(-2, 0.02^2)$ before softplus and the size floor. Membership weight logits and gene gate logits start at zero. Query scales s and offsets b start at zero, so the initial query correction is zero. Decoder affine scales start at one and biases at zero. Additional correlation and high-expression loss terms in the implementation have zero weight in this configuration.

Training and checkpoint selection. Each partition is trained independently on one GPU. Training batches are shuffled and incomplete final batches are dropped; validation and test batches preserve all examples. Python, NumPy, and PyTorch random generators are seeded with 42. Early stopping monitors the validation objective. The test configuration loads the final saved checkpoint at termination, rather than the checkpoint with minimum validation loss, and applies its EMA parameters. AdamW uses the library defaults for arguments other than the learning rate. MSE and MAE losses average over both batch and output dimensions; the mean and standard deviation penalties average over genes. Batch standard deviations use the population convention without an additional variance stabilizer.

Inference and postprocessing. Training samples one of eight paired image transforms: identity and the seven PIL transpose operations, applied identically to spot and niche images. Evaluation uses only the original spot and niche images, with the standard resizing and preprocessing, and no test-time augmentation. In addition to the jointly trained decoder affine layer, the launcher enables a separate gene-wise postprocessing calibrator. It fits a centered positive scale and bias to training predictions and targets, with shrinkage 0.8 toward the identity. The raw regression scale is clipped to $[0.8, 1.25]$ before shrinkage; nonpositive slopes are replaced by one, and the variance denominator has floor 10^{-6}. The fitted map is retained only if validation $\mathrm{MSE} + 0.1\,\mathrm{MAE}$ does not increase; otherwise the identity is used for all genes. This decision uses validation data, not test targets. Per-gene acceptance and distribution calibration are disabled. The retained affine map is applied to the gene predictions at test time. This postprocessing step is distinct from the decoder calibration ablation. The quoted external baseline results are not asserted to use this same inference pipeline.

We distinguish the model output $\hat{Y}_{ig}$ from the postprocessed evaluation prediction:

$$\hat{Y}_{ig}^{\mathrm{eval}} = a_g^{\mathrm{post}} \hat{Y}_{ig} + b_g^{\mathrm{post}}, \qquad a_g^{\mathrm{post}} > 0.$$

Here a_g^{post} and b_g^{post} are the fitted coefficients after clipping and shrinkage. If the fitted map is rejected by the validation criterion, $a_g^{\mathrm{post}} = 1$ and $b_g^{\mathrm{post}} = 0$ for all genes. This map is separate from the jointly learned decoder affine layer and is not part of the training objective $\mathcal{L}_{\mathrm{pred}}$.

C.3 Ablation definitions

For loss ablations, only the named coefficient is set to zero: $\lambda_{\mathrm{box}} = 0$ removes box supervision, $\lambda_{\mathrm{path}} = 0$ removes supervision of corrected pathway predictions while retaining $\mathcal{L}_{\mathrm{dir}}$, and $\lambda_{\mathrm{dir}} = 0$ removes direct pathway supervision while retaining $\mathcal{L}_{\mathrm{path}}$. Query removal sets the query correction to zero, leaving the direct pathway head. Decoder calibration removal uses the raw gated gene prediction throughout training. Disabling query or calibration is a change of the forward computation, rather than zeroing a learnable sigmoid logit or affine scale.

C.4 Pathway mask controls

All variants use the same five slide partitions within each tissue. Each variant is trained and evaluated separately on each partition. We summarize the resulting metrics by their mean and standard deviation across the five runs.

The gene label control applies a single permutation π to the columns of the real membership matrix, so $\widetilde{M}_{pg} = M_{p,\pi(g)}$, while leaving the ordered prediction gene list unchanged. This operation preserves pathway sizes, edge count, density, and the distribution of gene degrees, but generally changes the degree associated with each named gene. It also preserves pairwise pathway overlap under a common relabeling, so this control specifically disrupts the association between biological gene identity and pathway membership.

The degree preserving control uses seed 42 and $20E$ attempted edge swaps, where E is the number of memberships. Table 6 reports attempted and accepted swaps. For all four tissues, every pathway row sum and every gene column sum equal those of the corresponding real mask. Gene and pathway orders are unchanged in both controls.

For every gene label control, the row sums match the real mask, while the column sums for named genes change; the multiset of column sums remains the same. This distinction separates preservation

Table 6: Degree preserving pathway mask controls with seed 42 and $20E$ swap attempts. Row and column checks compare each degree with the real mask using the same gene and pathway order.

Tissue	Edges E	Attempts	Accepted	Row sums equal	Column sums equal
Kidney	1992	39840	26197	Yes	Yes
Colorectum	2085	41700	28140	Yes	Yes
Skin	2547	50940	24714	Yes	Yes
Lung	2248	44960	25594	Yes	Yes

of the degree distribution from preservation of each gene's degree. Kidney pathway score evaluation in Section 4.4 uses reference scores from the real mask for all models.

During training, the selected resource mask $M^{(v)}$ is shared by the point containment loss, the masked decoder, and both pathway supervision terms. Specifically, $A_i^{(v)} = Y_i(M_{\mathrm{norm}}^{(v)})^\top$ supervises both the direct and corrected pathway estimates. The ordered gene targets Y_i are unchanged. For the pathway analysis, reference scores are instead computed with the real mask for every variant. Thus, internal outputs of shuffled models represent randomized gene sets, and their agreement with real pathway scores assesses recovery of biological pathway semantics. Gene expression metrics compare all variants against the same targets.

C.5 MLP QUERY CONTROL AND PARAMETER COUNTS

The MLP control takes the same image representation as the query box branch and uses two residual MLP blocks followed by a linear pathway head. Its output replaces the overlap scores before the same learned pathway scale, bias, and sigmoid mixing coefficient are applied. The direct pathway predictor, both pathway supervision terms, masked pathway to gene mapping, direct gene branch, shared prediction losses, alignment, and calibration are retained. The control omits query and pathway boxes, gene points, and the geometric loss.

The hidden width is selected to minimize the absolute difference in trainable parameters used by the correction branch, without using validation or test performance. The Box reference includes the query encoder and output head, pathway centers and sizes, and the scale, bias, and mixing parameters. Gene points are excluded from this count because they enter geometric supervision rather than prediction. Table 7 gives counts derived from the model construction with a shared embedding width of 1,024, box dimension of 128, and MLP expansion ratio of 2. The absolute difference is below 0.1% for every tissue. These are branch counts rather than total model counts. The comparison controls prediction parameter count while also removing geometric supervision; the separate ablation without box loss evaluates the effect of that supervision within the Box model.

Table 7: Correction branch parameter counts derived from the model architecture. Relative difference is $(N_{\mathrm{MLP}} - N_{\mathrm{Box}})/N_{\mathrm{Box}}$. Counts include the learned scale, bias, and mixing coefficient in both branches and exclude gene points used only for geometric supervision.

Tissue	MLP hidden width	Box parameters	MLP parameters	Relative difference
Kidney	1,031	10,811,493	10,816,728	$+0.0484\%$
Colorectum	1,030	10,813,041	10,804,199	-0.0818%
Skin	1,028	10,825,683	10,817,300	-0.0774%
Lung	1,029	10,822,587	10,823,666	$+0.0100\%$

D ADDITIONAL RESULTS AND VISUALIZATIONS

This appendix reports the component ablations and pathway mask controls for Colorectum, Skin, and Lung. Kidney results are reported in Tables 2 and 3.

It also presents pathway score agreement and spatial visualizations following the analyses in Section 4.4.

D.1 COLORECTUM ABLATIONS AND PATHWAY MASK CONTROLS

Tables 8 and 9 report the component ablations and pathway mask controls on Colorectum.

Table 8: Component ablations on Colorectum. Values are means with standard deviations. Bold and underline indicate the best and second-best means in each column, including ties at the displayed precision.

Variant	PCC@200 ↑	PCC@50 ↑	PCC@10 ↑	MSE ↓	MAE ↓
HyBoxST	$\mathbf{0.438 \pm 0.170}$	$\mathbf{0.646 \pm 0.136}$	$\mathbf{0.734 \pm 0.104}$	$\mathbf{1.481 \pm 0.474}$	$\mathbf{0.941 \pm 0.164}$
w/o query box	0.414 ± 0.180	0.601 ± 0.154	0.697 ± 0.119	1.525 ± 0.596	0.970 ± 0.192
MLP query	0.411 ± 0.166	0.609 ± 0.145	0.704 ± 0.114	1.496 ± 0.488	0.956 ± 0.162
w/o box loss	0.417 ± 0.167	0.615 ± 0.146	0.712 ± 0.116	$\underline{1.486 \pm 0.492}$	0.958 ± 0.167
w/o pathway loss	0.419 ± 0.164	0.617 ± 0.149	0.716 ± 0.117	1.494 ± 0.487	0.962 ± 0.162
w/o direct pathway loss	0.432 ± 0.168	0.634 ± 0.147	0.722 ± 0.113	1.492 ± 0.489	$\underline{0.943 \pm 0.163}$
w/o decoder calibration	$\underline{0.436 \pm 0.169}$	$\underline{0.643 \pm 0.152}$	$\underline{0.732 \pm 0.114}$	1.540 ± 0.521	0.967 ± 0.180

Table 9: Pathway mask controls on Colorectum. Values are means with standard deviations. Bold and underline indicate the best and second-best means in each column, including ties at the displayed precision.

Mask setting	PCC@200 ↑	PCC@50 ↑	PCC@10 ↑	MSE ↓	MAE ↓
Real pathway mask	$\mathbf{0.438 \pm 0.170}$	$\mathbf{0.646 \pm 0.136}$	$\mathbf{0.734 \pm 0.104}$	$\mathbf{1.481 \pm 0.474}$	$\mathbf{0.941 \pm 0.164}$
Shuffled mask	0.404 ± 0.172	0.618 ± 0.137	0.706 ± 0.105	1.521 ± 0.535	0.952 ± 0.176
Degree preserving shuffle	$\underline{0.415 \pm 0.170}$	$\underline{0.622 \pm 0.146}$	$\underline{0.711 \pm 0.115}$	$\underline{1.517 \pm 0.546}$	$\underline{0.948 \pm 0.177}$

On Colorectum, the full model attains the best mean on each of the five metrics in the component ablation table. Removing the query box reduces PCC@200 from 0.438 to 0.414 and increases MSE from 1.481 to 1.525. The differences for some other components are smaller: removing the box loss gives MSE of 1.486, while removing the direct pathway loss gives PCC@200 of 0.432 and MAE of 0.943, compared with 0.438 and 0.941 for the full model. The real pathway mask also outperforms both shuffled controls on all five metrics. These comparisons describe the reported means without establishing statistical significance.

The MLP query control obtains PCC@200 of 0.411, MSE of 1.496, and MAE of 0.956. Compared with removing the query branch, it has slightly lower PCC@200 (0.411 versus 0.414), but lower MSE (1.496 versus 1.525) and MAE (0.956 versus 0.970). The full model retains better means on all five metrics than the MLP replacement.

D.2 SKIN ABLATIONS AND PATHWAY MASK CONTROLS

Tables 10 and 11 report the component ablations and pathway mask controls on Skin.

Table 10: Component ablations on Skin. Values are means with standard deviations. Bold and underline indicate the best and second-best means in each column, including ties at the displayed precision.

Variant	PCC@200 ↑	PCC@50 ↑	PCC@10 ↑	MSE ↓	MAE ↓
HyBoxST	$\mathbf{0.803 \pm 0.072}$	$\mathbf{0.857 \pm 0.060}$	$\mathbf{0.876 \pm 0.052}$	$\mathbf{0.927 \pm 0.130}$	$\mathbf{0.661 \pm 0.078}$
w/o query box	0.762 ± 0.084	0.822 ± 0.072	0.844 ± 0.059	0.950 ± 0.135	0.674 ± 0.079
MLP query	0.767 ± 0.075	0.823 ± 0.061	0.845 ± 0.053	0.940 ± 0.128	0.674 ± 0.071
w/o box loss	0.773 ± 0.078	0.831 ± 0.069	0.852 ± 0.056	0.940 ± 0.134	$\underline{0.670 \pm 0.077}$
w/o pathway loss	0.780 ± 0.076	0.838 ± 0.076	0.860 ± 0.058	0.942 ± 0.126	0.679 ± 0.073
w/o direct pathway loss	0.784 ± 0.076	0.841 ± 0.061	$\underline{0.862 \pm 0.052}$	$\underline{0.939 \pm 0.127}$	0.672 ± 0.070
w/o decoder calibration	$\underline{0.801 \pm 0.077}$	$\underline{0.854 \pm 0.064}$	$\mathbf{0.876 \pm 0.055}$	0.942 ± 0.129	0.680 ± 0.075

On Skin, the full model has the best values across all five metrics in the component ablation table. Removing the query box reduces PCC@200 from 0.803 to 0.762. The shuffled mask and degree

Table 11: Pathway mask controls on Skin. Values are means with standard deviations. Bold and underline indicate the best and second-best means in each column, including ties at the displayed precision.

Mask setting	PCC@200 ↑	PCC@50 ↑	PCC@10 ↑	MSE ↓	MAE ↓
Real pathway mask	$\mathbf{0.803 \pm 0.072}$	$\mathbf{0.857 \pm 0.060}$	$\mathbf{0.876 \pm 0.052}$	$\mathbf{0.927 \pm 0.130}$	$\mathbf{0.661 \pm 0.078}$
Shuffled mask	0.778 ± 0.076	0.833 ± 0.067	0.852 ± 0.054	0.949 ± 0.129	0.672 ± 0.074
Degree preserving shuffle	$\underline{0.788 \pm 0.079}$	$\underline{0.839 \pm 0.065}$	$\underline{0.864 \pm 0.057}$	$\underline{0.942 \pm 0.129}$	$\underline{0.670 \pm 0.071}$

preserving control obtain PCC@200 of 0.778 and 0.788, respectively, and both have higher MSE and MAE than the real mask.

With the MLP query replacement, PCC@200 is 0.767 and MSE is 0.940, compared with 0.762 and 0.950 when the query branch is removed. The full model reaches PCC@200 of 0.803 and MSE of 0.927, with better means on all five metrics than the MLP control.

D.3 Lung ablations and pathway mask controls

Tables 12 and 13 report the component ablations and pathway mask controls on Lung.

Table 12: Component ablations on Lung. Values are means with standard deviations. Bold and underline indicate the best and second-best means in each column, including ties at the displayed precision.

Variant	PCC@200 ↑	PCC@50 ↑	PCC@10 ↑	MSE ↓	MAE ↓
HyBoxST	$\mathbf{0.592 \pm 0.160}$	$\mathbf{0.752 \pm 0.131}$	$\mathbf{0.819 \pm 0.083}$	$\mathbf{0.811 \pm 0.409}$	$\mathbf{0.602 \pm 0.186}$
w/o query box	0.538 ± 0.169	0.717 ± 0.147	0.771 ± 0.113	0.845 ± 0.414	0.628 ± 0.184
MLP query	0.552 ± 0.160	0.717 ± 0.131	0.785 ± 0.083	0.842 ± 0.409	0.625 ± 0.186
w/o box loss	0.560 ± 0.164	0.724 ± 0.142	0.793 ± 0.083	0.836 ± 0.411	$\underline{0.623 \pm 0.186}$
w/o pathway loss	0.563 ± 0.167	0.727 ± 0.145	0.799 ± 0.081	0.832 ± 0.413	0.625 ± 0.185
w/o direct pathway loss	0.579 ± 0.165	0.736 ± 0.137	0.808 ± 0.082	$\underline{0.830 \pm 0.414}$	0.624 ± 0.188
w/o decoder calibration	$\underline{0.589 \pm 0.170}$	$\underline{0.750 \pm 0.139}$	$\underline{0.816 \pm 0.083}$	0.836 ± 0.406	0.629 ± 0.190

Table 13: Pathway mask controls on Lung. Values are means with standard deviations. Bold and underline indicate the best and second-best means in each column, including ties at the displayed precision.

Mask setting	PCC@200 ↑	PCC@50 ↑	PCC@10 ↑	MSE ↓	MAE ↓
Real pathway mask	$\mathbf{0.592 \pm 0.160}$	$\mathbf{0.752 \pm 0.131}$	$\mathbf{0.819 \pm 0.083}$	$\mathbf{0.811 \pm 0.409}$	$\mathbf{0.602 \pm 0.186}$
Shuffled mask	0.565 ± 0.165	0.726 ± 0.135	0.793 ± 0.086	0.839 ± 0.403	0.632 ± 0.184
Degree preserving shuffle	$\underline{0.571 \pm 0.169}$	$\underline{0.733 \pm 0.142}$	$\underline{0.804 \pm 0.095}$	$\underline{0.832 \pm 0.398}$	$\underline{0.627 \pm 0.180}$

On Lung, the full model has the best values across all five metrics in the component ablation table. Removing the query box reduces PCC@200 from 0.592 to 0.538 and increases MAE from 0.602 to 0.628. The shuffled mask and degree preserving control obtain PCC@200 of 0.565 and 0.571, respectively, and both have higher MSE and MAE than the real mask.

The MLP query control obtains PCC@200 of 0.552, MSE of 0.842, and MAE of 0.625. Relative to this control, the full model increases PCC@200 to 0.592 and reduces MSE and MAE to 0.811 and 0.602. As on the other tissues, its means are better on all five metrics than those of the MLP replacement.

D.4 Interpretation of component ablations

Removing the query correction causes the largest decrease in all three PCC metrics among the Kidney ablations, with PCC@200 falling from 0.428 to 0.402. The MLP correction yields PCC@200 of 0.403 and MSE of 1.127, compared with 0.402 and 1.139 without a correction branch. The full model reaches 0.428 and 1.020, respectively, and has better means on all five metrics than the MLP control. Removing the query also worsens all five metrics in each of the other tissues (Appendix D).

To separate the two changes involved in replacing the MLP with the full model, we first compare the MLP with box queries trained without membership loss, then add membership supervision. Across Kidney, Colorectum, Skin, and Lung, the first comparison increases mean PCC@200 by 0.013, 0.006, 0.006, and 0.008, and the second by 0.012, 0.021, 0.030, and 0.032. These results support contributions from both query structure and membership supervision within the tested configurations. The first comparison does not improve every metric: Colorectum MAE is 0.958 versus 0.956 for the MLP, and Skin MSE ties at 0.940. These descriptive comparisons do not establish statistical significance or superiority over other representations.

The remaining ablations examine supervision of pathway scores and expression scale. On Kidney, removing either corrected or direct pathway supervision lowers all three mean correlations; removing direct supervision also increases both errors. Training without decoder calibration mainly increases the errors. Across all four tissues, the full model has the best means in the component tables, including ties at the displayed precision (Appendix D).

D.5 Pathway expression score alignment

We next assess whether the pathway signals used by the decoder agree with observed member expression on Kidney. We evaluate both internal estimates P_i and scores aggregated from final gene predictions against reference means defined by real memberships (Figure 3). These expression means differ from ssGSEA scores and other measures of functional coherence (Majumder et al., 2026; Hu et al., 2026). Shuffled models are also evaluated against the real references, although their internal outputs were trained on randomized gene sets. Their comparison therefore measures correspondence to real pathway scores, rather than isolating a mechanism advantage on a shared pathway training target; the gene prediction controls provide the comparison with a common target.

PCC and MAE are computed by pooling spot and pathway predictions across five runs, whereas gene metrics are averaged over runs. The pooled values describe overall score agreement: they combine differences between pathways and slides with variation within pathways, and test slides may recur across runs. They therefore do not isolate spatial agreement within individual pathways or represent five disjoint test sets.

Internal pathway PCC/MAE are 0.673/0.580 for the real mask, 0.274/0.804 for gene label shuffling, and 0.478/0.724 for degree preserving shuffling. For scores aggregated from gene predictions, the corresponding pairs are 0.691/0.493, 0.598/0.586, and 0.612/0.536. The real mask has the highest pooled PCC and lowest pooled MAE at both stages, indicating closer overall agreement with the real pathway references.

D.6 Spatial visualization of predicted gene expression

Figure 4 compares spatial expression maps for UMOD and PODXL on the NCBI704 section. HyBoxST achieves PCC values of 0.857 and 0.819 for these genes, respectively, compared with 0.724 and 0.692 for HyperST. The figure also includes StNet, BLEEP, Stem, and TRIPLEX. These examples illustrate agreement between predicted and observed expression patterns on this section; the dataset results in Table 1 provide the broader quantitative comparison.

D.7 Query contributions for additional pathways

Main example and evaluation. For the main spatial example, Figure 2 decomposes Metabolism of lipids predictions on NCBI704 into P_{ip}^{dir}, $\Delta_{ip}^{\mathrm{query}} = \sigma(\eta)(s_p o_{ip} + b_p)$, and their sum P_{ip}. The reference A_{ip} averages five measured member genes after the transformation in Equation 8. The spotwise error reduction is $(A_{ip} - P_{ip}^{\mathrm{dir}})^2 - (A_{ip} - P_{ip})^2$, with positive values indicating improvement. All branches come from the same trained model, so this analysis complements, rather than replaces, retrained ablations.

The correction raises PCC from 0.491 to 0.627 and lowers MSE from 0.2546 to 0.2275 (10.6%), although some spots worsen. Two further examples are shown below. These measured member scores reveal how the query affects predictions, but do not establish activity of an entire biological process.

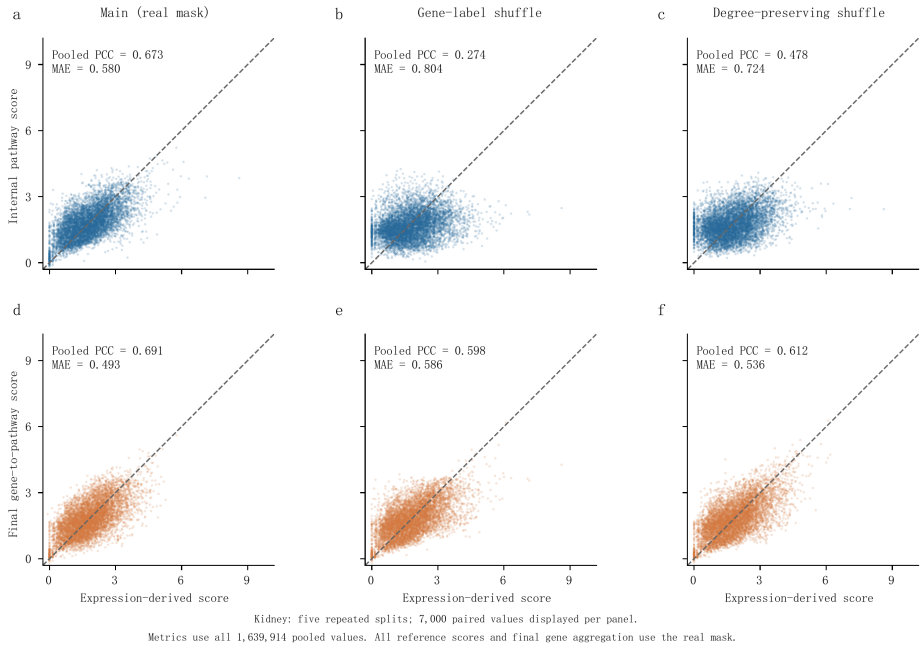

Figure 3: Pathway score alignment on Kidney across five runs. Columns compare the real mask, gene label shuffle, and degree preserving shuffle. Rows evaluate internal pathway scores and scores aggregated from final gene predictions. All reference scores use the real mask. Metrics use all pooled values; 7,000 points are shown per panel. Dashed lines indicate equal prediction and reference values.

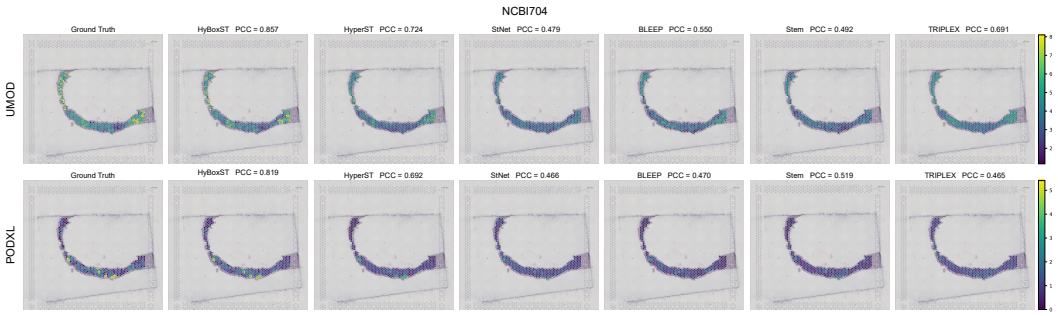

Figure 4: Spatial expression of UMOD (top) and PODXL (bottom) on NCBI704. Columns show observed expression and predictions from HyBoxST, HyperST, StNet, BLEEP, Stem, and TRIPLEX. PCC is reported above each prediction map.

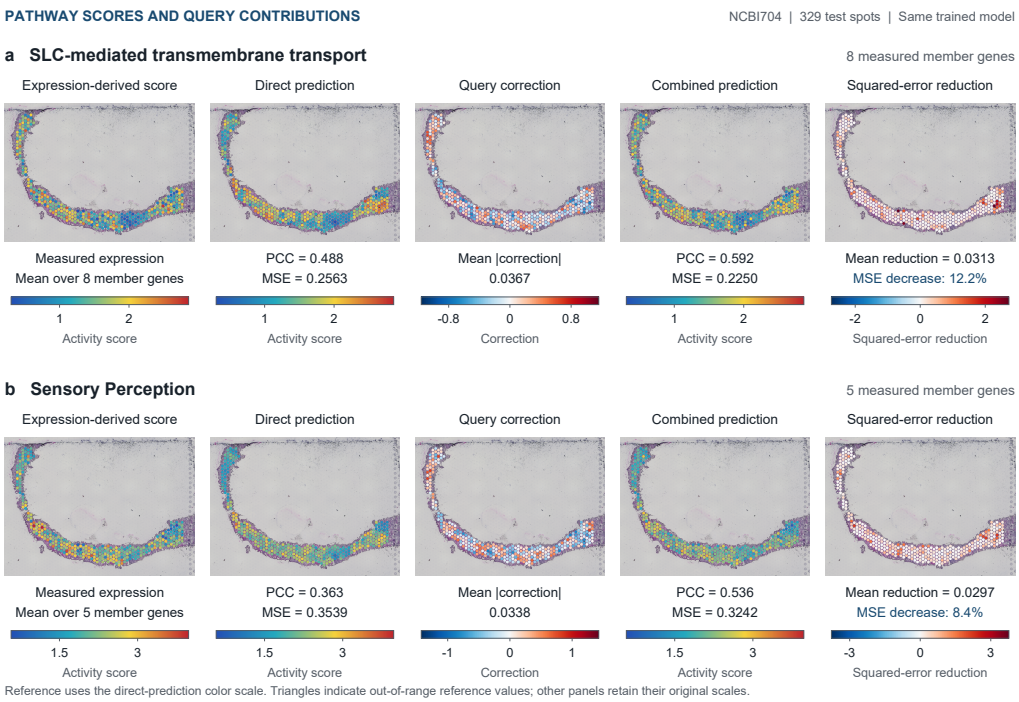

Figure 5: Query corrections for two additional pathways on NCBI704 (329 test spots; the same trained model as Figure 2). Rows show SLC mediated transmembrane transport with eight measured member genes and Sensory Perception with five. Columns show the expression derived reference, direct prediction, query correction, combined prediction, and squared error reduction. Metrics and color meanings follow Figure 2. Within each row, reference and direct prediction share a color scale; the combined prediction retains its displayed scale. Query correction and squared error reduction use separate signed scales. "Activity score" denotes the pathway expression score. Colorbar triangles mark reference values beyond the displayed limits.

Figure 5 extends the decomposition in Figure 2 to SLC mediated transmembrane transport and Sensory Perception, using the same trained model and the same 329 NCBI704 test spots. Their reference scores average measured expression after the $\log_2(x+1)$ transformation over eight and five member genes, respectively. The correction and squared error reduction follow the definitions in Section 4.4.

For SLC mediated transmembrane transport, adding the query correction increases PCC from 0.488 to 0.592 and decreases MSE from 0.2563 to 0.2250, a 12.2% reduction. For Sensory Perception, PCC increases from 0.363 to 0.536 and MSE decreases from 0.3539 to 0.3242, an 8.4% reduction. Mean absolute correction magnitudes are 0.0367 and 0.0338, and mean squared error reductions are 0.0313 and 0.0297, respectively. Both maps contain improvements and deteriorations at individual spots. Together with the main text example, these results illustrate how the query branch changes intermediate predictions within a fixed model; they do not replace retrained ablations or establish uniform benefits across pathways and tissues.