Privatized Barker's Descent Algorithm

CS690 (Computational Genomics)

Supervisor: Prof. Hamim Zafar
(Group2) Zehaan Naik, Aditya V

Dept. of Mathematics and Statistics
Indian Institute of Technology Kanpur

Nov, 2025

Problem Statement

Xenium Annotation

- 10x Genomics **Xenium** provides high-resolution spatial gene expression, but uses **targeted panels** with limited gene coverage.
- Xenium profiles are **sparse, low-count**, and not directly comparable to high-coverage **scRNA-seq reference datasets**.
- Existing annotation tools assume **full-transcriptome** data, leading to poor transfer performance on Xenium cells (Cheng et al., 2025).
- Goal: **accurately annotate cell types** in Xenium by learning a **shared latent representation** that is:
  - aligned across modalities (scRNA-seq and Xenium), and
  - preserves biological cell-type structure.
- Challenge: integrate **spatial information** from Xenium while preventing over-alignment and maintaining cell-type separability.

Background I

Current State-of-the-art

- **Spatial transcriptomics** (e.g., Xenium) enables the measurement of subcellular, in-situ gene expression — a powerful tool for studying tissue organization, but presents unique challenges (Cheng et al., 2025).
- **Key technical constraints of Xenium:**
  - Targeted gene panels $\rightarrow$ *limited gene coverage*.
  - Low per-cell counts $\rightarrow$ *sparse, noisy profiles*.
  - Strong *spatial* information that can complement expression data.
- **Why off-the-shelf scRNA-seq methods fail:**
  - Many annotation tools assume full-transcriptome, high-coverage measurements.
  - Direct label transfer often propagates modality-specific biases and loses subtype resolution.

Background II

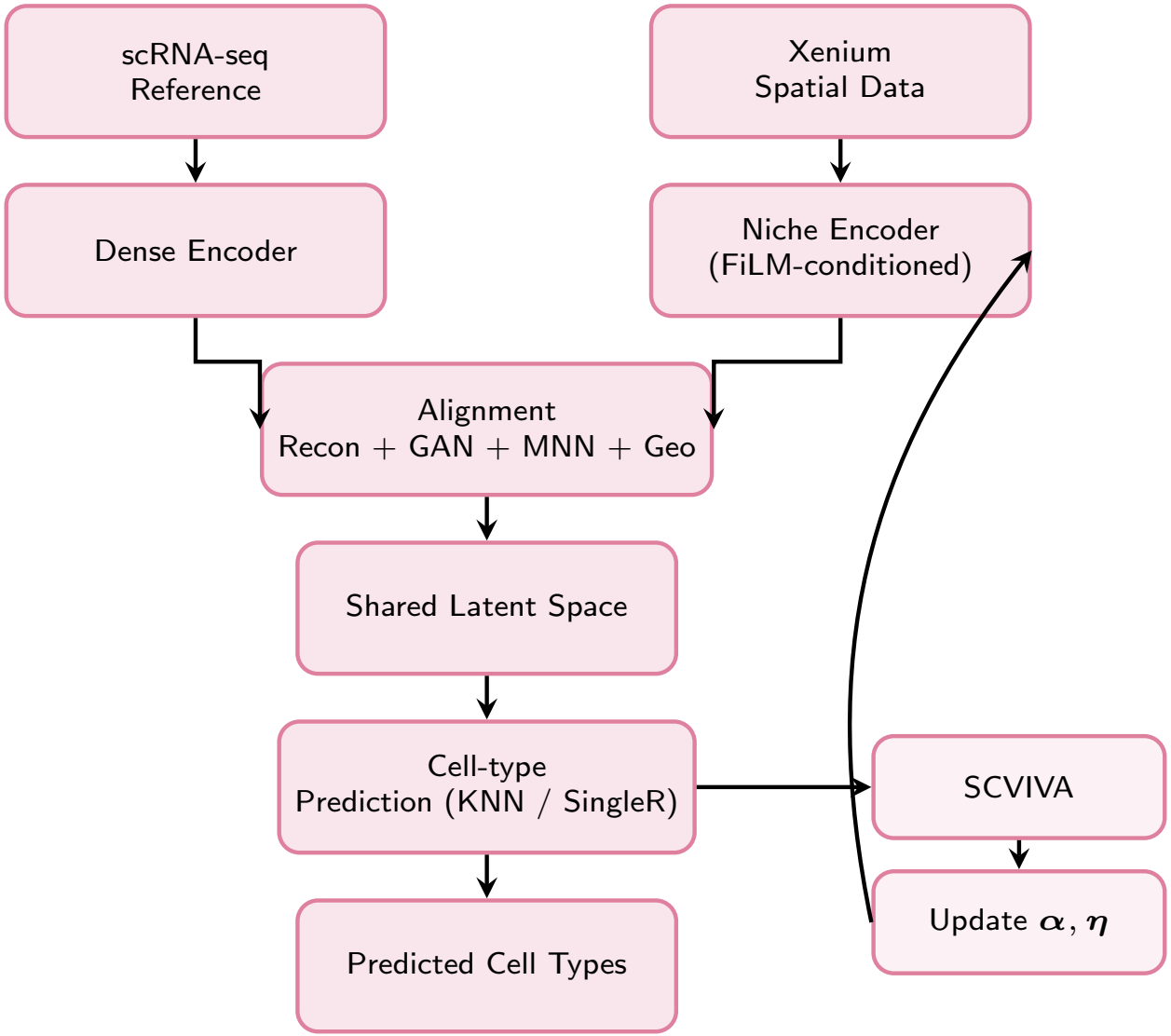
- **Relevant prior approaches (high-level):**
  - **CellSymphony** (Acosta et al., 2024): positional encodings within transformer-style models to bring spatial context into prediction pipelines, but spatial info is not used to directly drive cross-modal alignment.
  - **Non-parametric label-transfer (e.g., SingleR, Azimuth)** (Cheng et al., 2025): effective when datasets are similar, but less reliable under modality/resolution mismatch.
  - **SCVIVA** (Zhang et al., 2023): explicitly models niche/context to augment VAE-based latent spaces, showing the benefit of neighborhood (niche) structure.
  - **SCModal** (Chavez et al., 2022): dual-autoencoder + adversarial alignment for cross-modal embedding — a useful foundation for further spatial extensions.
- **Takeaway:** Spatial context and careful alignment strategies are both necessary — naive alignment risks over-mixing; preserving cell-type discriminability is critical.

  Next: a focused discussion of *SCArches*-based latent alignment and how it informs our adaptation strategy (next slide).

Proposed Method: Overview

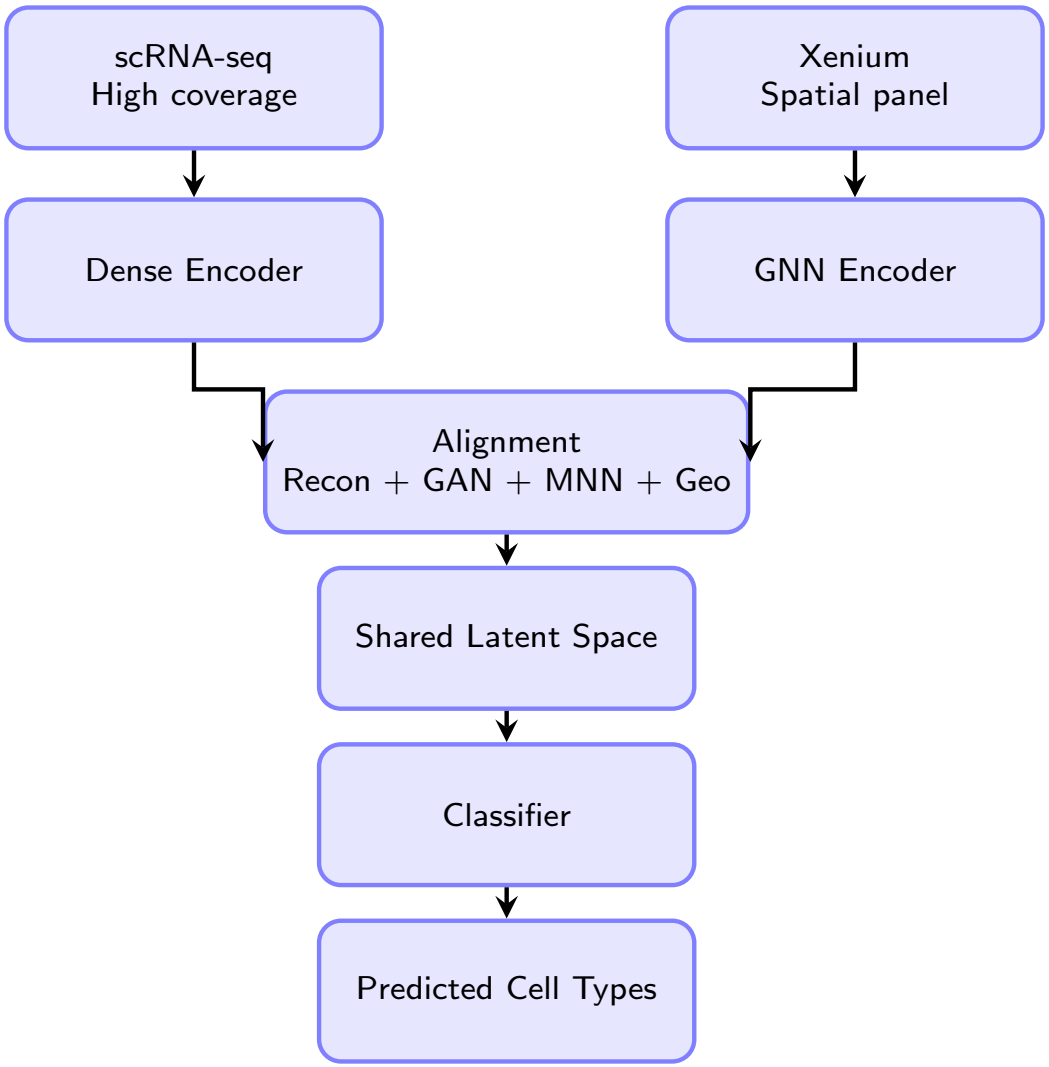
- **Objective:** Learn a shared latent space where:
  - Xenium spatial profiles and scRNA-seq reference profiles are aligned
  - Biologically meaningful cell-type structure is preserved.
- **Motivation:**
  - Xenium captures strong spatial structure but limited gene expression.
  - SCRNA-seq captures rich transcriptional variation but lacks spatial context.
  - **Combining both modalities** strengthens annotation robustness.
- **Model Foundation:** We extend the SCModal (Chavez et al., 2022) dual-autoencoder framework with:
  - 1 An iterative niche conditioned encoder
  - 2 A **spatially-aware GNN encoder** for Xenium
  - **Modality alignment losses** (reconstruction, cross-reconstruction, adversarial, geometric),
  - **MNN-based local correspondence** reinforcement,
  - A **supervised classification loss** to preserve cell-type separability.
- **High-level pipeline:**
  1. Preprocess and gene-match scRNA-seq and Xenium data.
  2. Encode scRNA-seq with a dense encoder; encode Xenium with a GNN.
  3. Align latent spaces using modified SCModal losses.
  4. Train a classifier on reference embeddings.
  5. Transfer predicted labels to Xenium.

Graphical Abstract — Niche-SCModal



Niche-SCModal includes an iterative SCVIVA-driven niche update loop that refines FiLM conditioning across training iterations.

Graphical Abstract



A compact overview of our Xenium annotation workflow.

Preprocessing and Notation

- **Datasets**
  - $X_A \in \mathbb{R}^{n_A \times p_A}$: scRNA-seq reference gene expression matrix.
  - $X_B \in \mathbb{R}^{n_B \times p_B}$: Xenium spatial gene expression matrix.
  - $C_B \in \mathbb{R}^{n_B \times d}$: Spatial coordinates for Xenium cells ($d = 2$).
- **Latent representation**
  - $z_{A,i}, z_{B,j} \in \mathbb{R}^q$: latent embeddings for cell i (reference) and cell j (Xenium).
  - q is the shared latent dimensionality (e.g., 32–64).
- **Graphs and neighborhoods**
  - Construct spatial kNN graph on coordinates C_B for GNN message passing.
  - Build shared-gene kNN graph (minibatch-wise) for MNN correspondence.
- **Model components (to be detailed next)**
  - Encoders: E_A (GNN for Xenium), E_B (dense encoder for scRNA-seq).
  - Generators/decoders: G_A and G_B.
  - Latent-space discriminator D_Z and classifier C.

Spatial GNN Encoder for Xenium I

Goal

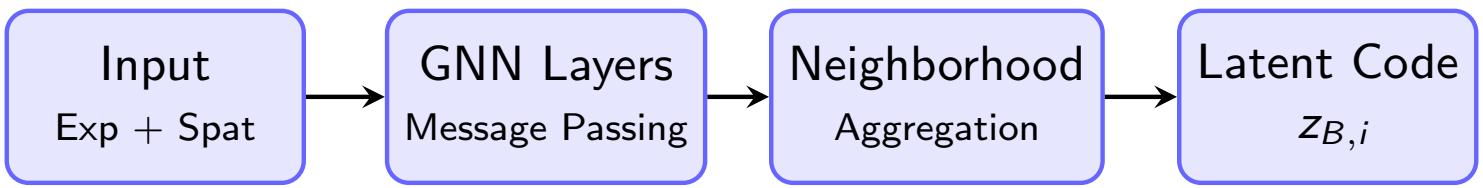
Incorporate spatial organization of Xenium cells directly into the encoding process.

Key idea

Each cell's representation is influenced not only by its own expression, but also by its *spatial neighbors*.

Graph Construction

- Build a kNN graph on coordinates C_B.
- Nodes: Xenium cells; Edges: spatial neighbors.



Spatial GNN Encoder for Xenium II

- **GNN Architecture**
  - Node features: Concatenation of expression vector $x_{B,j}$ and a low-dimensional coordinate embedding.
  - Apply L message-passing layers:

$$h_i^{(\ell+1)} = \sigma\left(W^{(\ell)} h_i^{(\ell)} + \sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(\ell)} U^{(\ell)} h_j^{(\ell)} \right)$$

  - Final latent embedding:

$$z_{B,i} = \mathrm{Proj}\left(h_i^{(L)}\right) \in \mathbb{R}^q$$

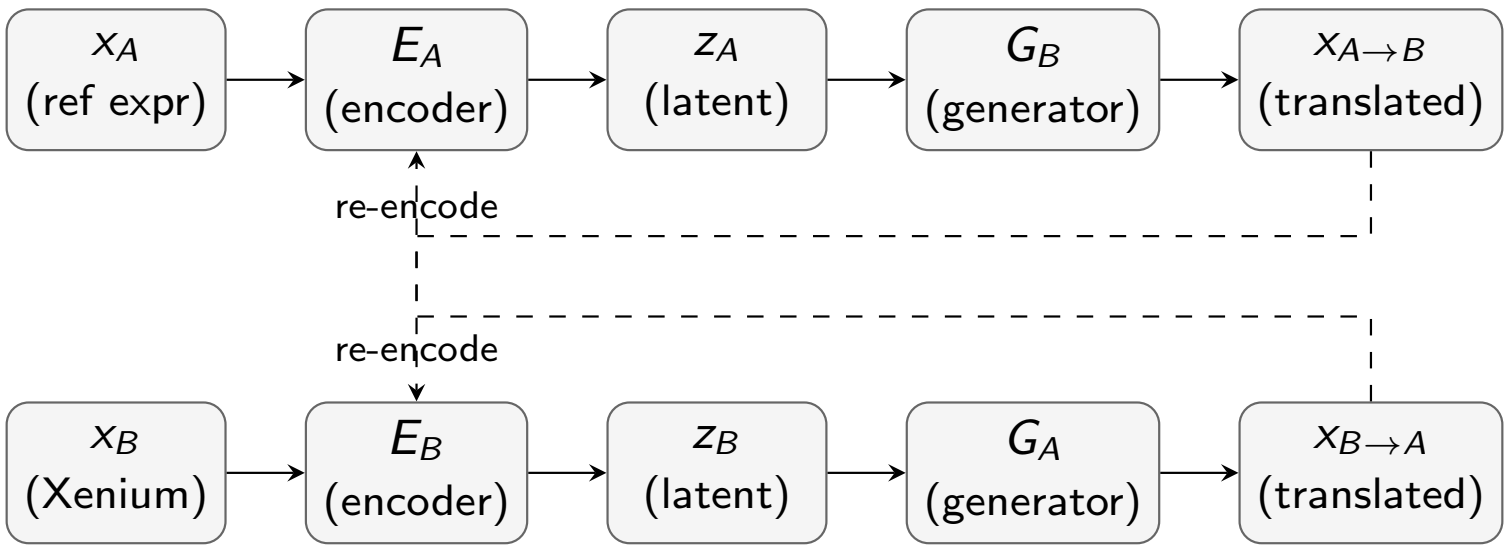
- **Benefits**
  - Embeddings reflect both local niche structure and gene expression.
  - Prevents biologically implausible mixing during alignment.
  - Provides spatial inductive bias missing in SCModal and other methods.

Generators / Decoders & Cross-Modal Translation I

- **Generators / Decoders**
  - Each modality has its own generator that maps latent vectors back to input space:
$$\hat{x}_{A,i} = G_A(z_{A,i}), \qquad \hat{x}_{B,j} = G_B(z_{B,j}).$$
  - Generators implemented as two-layer feedforward networks (ReLU + linear output).
- **Cross-domain translation (intuition)**
  - Translate a latent code from one modality into the other modality's input space using the corresponding generator, then re-encode to check consistency.
  - Encourages that an encoding from one domain maps to a semantically similar code in the other domain.
- **Cross-reconstruction / Latent-alignment**
  - For x_A (reference): compute $z_A = E_A(x_A)$, translate $x_{A\to B} = G_B(z_A)$, then re-encode $z_{A\to B} = E_B(x_{A\to B})$.
  - Penalize mismatch:
$$L_{\text{LA}} \supset \|z_A - z_{A\to B}\|_2^2 \quad \text{(and the symmetric term for } x_B\text{)}.$$
  - This enforces cross-modal consistency beyond simple reconstruction.

Generators / Decoders & Cross-Modal Translation II

Cross-domain translation



Generators / Decoders & Cross-Modal Translation III

- **Mutual Nearest Neighbors (MNN) for sparse correspondences**
  - Compute batch-wise mutual nearest neighbors in the shared-gene subspace; form binary matrix S with $S_{ij} = 1$ for MNN pairs.
  - MNN loss penalizes latent distance for matched pairs:

$$L_{\text{MNN}} = \frac{\sum_{i,j} S_{ij} \|z_{A,i} - z_{B,j}\|_2^2}{\sum_{i,j} S_{ij} + \varepsilon}.$$

  - Reinforces local correspondences and helps avoid global mode collapse.

Practical notes:

- We used approximate nearest neighbors (Annoy / sklearn) for MNN in minibatches.
- When translating spatial $\rightarrow$ reference, we provided an appropriate coordinate input (original or zero) when required by the GNN encoder.

Results: Overview

Objective of the Results Section

- Evaluate how spatial information and model architecture choices influence cross-modal alignment and cell-type annotation accuracy.
- Compare three major experimental settings:
  1. Niche-SCModal (FiLM-based spatial conditioning)
  2. Niche-SCModal + SingleR label refinement
  3. Graph-SCModal (GNN-based spatial encoder)

Key Questions We Investigate

- Does explicit spatial conditioning improve alignment stability?
- How well do different classifiers (KNN vs SingleR) transfer labels?
- What are the trade-offs in ARI, F1, and per-class performance?

What the Results Will Show

- Quantitative comparisons: ARI, accuracy, F1-scores across cell types.
- UMAP visualizations of latent spaces for each model variant.

Niche-SCModal: Results

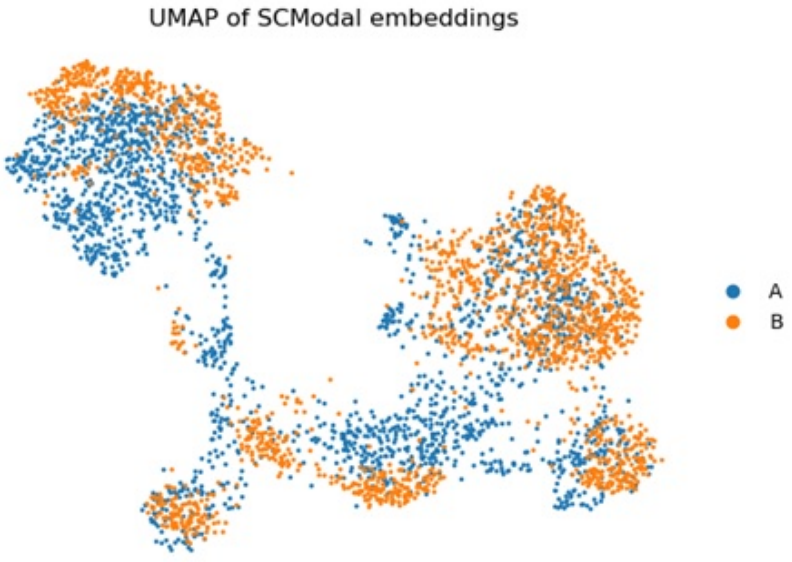


Figure: Niche-SCModal latent space UMAP (reference + Xenium).

ARI comparison

Model	ARI
Niche-SCModal	0.5732
Base SCModal	0.5162

Key observations

- Niche conditioning (FiLM-based) consistently improves ARI by ≈5 points on the full dataset.
- Improvements scale with dataset size — spatial conditioning helps more on larger datasets.
- Niche-SCModal preserves local neighborhood structure while modestly improving alignment.
- Practical note: niche guidance is applied via iterative KNN-based label updates during training.

Graph-SCModal: Results

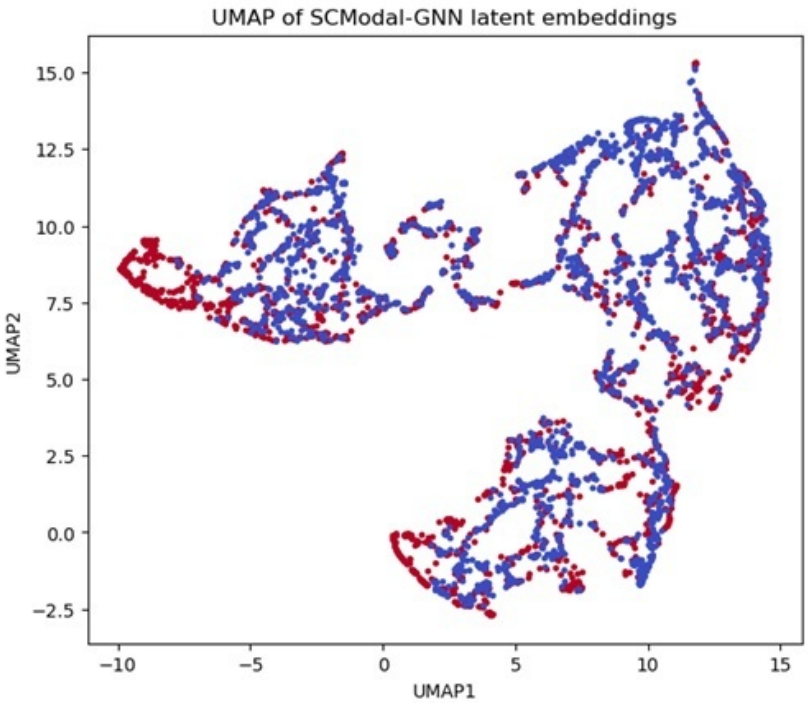


Figure: Graph-SCModal latent space UMAP (subset of 2000 cells).

Performance summary (subset)

Metric	Score
Accuracy	0.57
ARI	0.53

Class-wise F1-scores (selected):

- Cancer Epithelial: 0.91
- CAFs: 0.60
- T-cells: 0.44
- Endothelial: 0.14
- Normal Epithelial: 0.53

Key observations

- On larger datasets, the model exhibits *over-alignment* and partial collapse of biologically meaningful structure.
- Indicates the need for improved balancing of GAN, geometric, and MNN losses.

Niche-SCModal: SingleR vs KNN Label Transfer

F1-score comparison

Cell Type	SingleR	KNN
B-cells	0.45	0.06
CAFs	0.80	0.66
Cancer Epithelial	0.88	0.92
Endothelial	0.75	0.83
Myeloid	0.78	0.74
Normal Epithelial	0.33	0.63
T-cells	0.75	0.57
Accuracy	0.72	0.70

Key observations

- SingleR improves classification of several immune subtypes, especially lymphocytes (B- and T-cells).
- KNN remains competitive for major epithelial populations.
- Overall accuracy remains comparable (0.72 vs 0.70), but SingleR yields more biologically faithful immune labeling.

SCArches

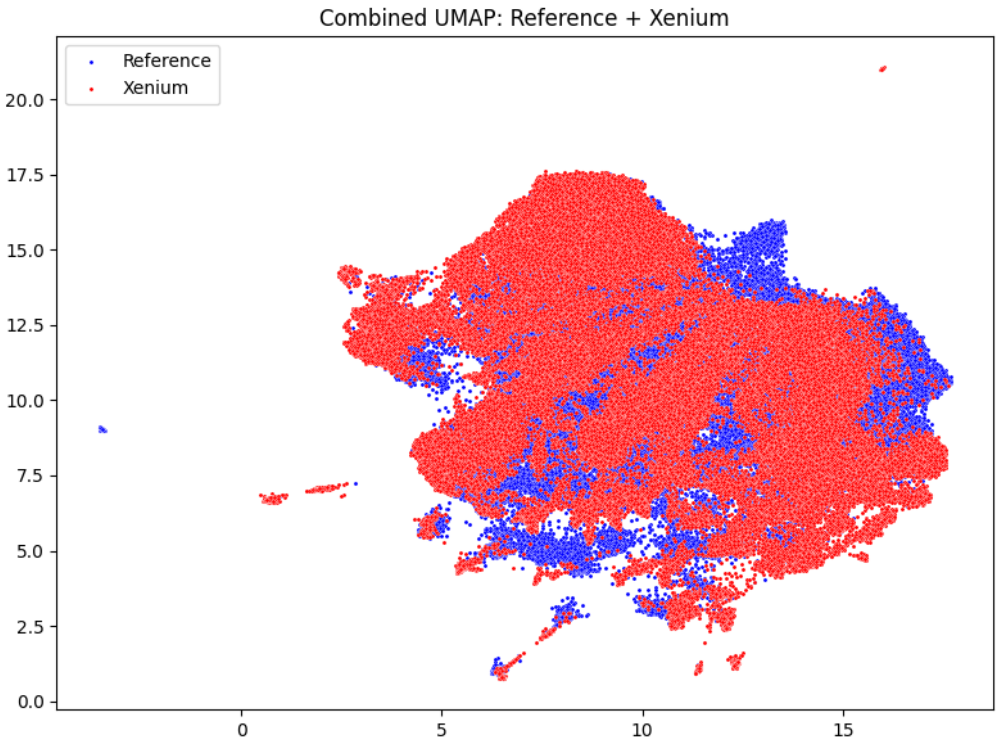
Method-I

SCArches (**Single-Cell Architectural Surgery**) (Lotfollahi et al., 2022) is a transfer-learning framework. We implement SCArches for our problem statement and discuss the results thus obtained.

Latent structure:

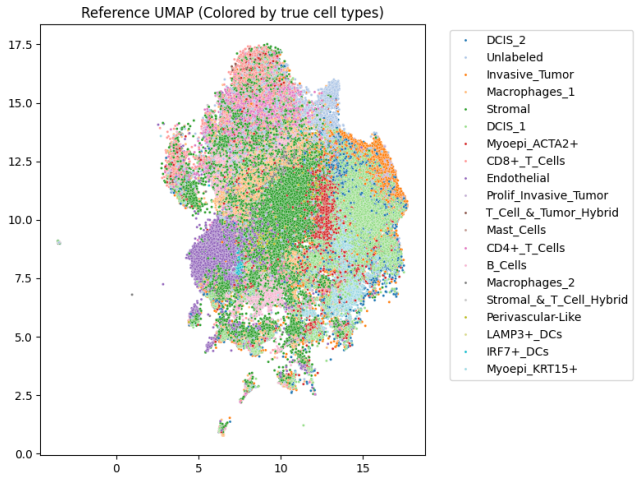
- Silhouette (Reference): 0.0491; Silhouette (Xenium): 0.0539.
- High ARI on reference $\rightarrow$ latent geometry preserves biological structure.
- Xenium ARI indicates reasonable alignment, but lower accuracy suggests modality gaps / limited gene coverage.
- Visuals: good co-localization in combined UMAP; some class mixing expected due to targeted panel sparsity.

SCArches: Results

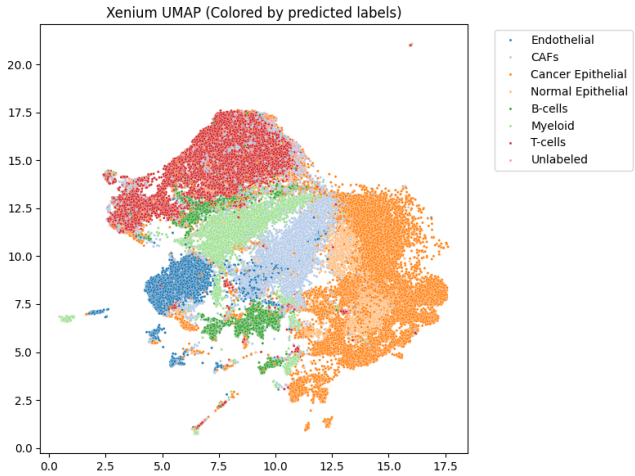


Quantitative metrics

Dataset	Accuracy	ARI
Reference	0.9016	0.8396
Query	0.4975	0.6856



(a) Reference labels



(b) Xenium predicted labels

Conclusion

- **Spatial information matters:** Niche-conditioning and GNN-based encoders consistently improve alignment stability and preserve biologically meaningful structure in Xenium data.
- **Model choice affects different cell types:** Spatial models boost epithelial and CAF predictions, while SingleR enhances lymphocyte identification; no single classifier dominates across all classes.
- **Graph-SCModal shows strongest spatial sensitivity:** GNN encoders capture local tissue topology effectively, though careful loss balancing is required to avoid over-alignment on larger datasets.

Overall: Integrating spatial context with cross-modal alignment substantially improves cell-type annotation for targeted in situ transcriptomics.

Thank you!

References I

Acosta, P. H., Chen, P., Castillo, S. P., Salvatierra, M. E., Yuan, Y., and Pan, X. (2024). Cellsymphony: Deciphering the molecular and phenotypic orchestration of cells with single-cell pathomics. *Nature Communications*. In press.

Chavez, S., Li, M., Patel, R., and Singh, A. (2022). Scmodal: Cross-modal integration of single-cell transcriptomics via dual autoencoders and adversarial alignment. *bioRxiv*.

Cheng, J., Jin, X., Smyth, G. K., et al. (2025). Benchmarking cell type annotation methods for 10x xenium spatial transcriptomics data. *BMC Bioinformatics*, 26:22.

Lotfollahi, M., Wolf, F. A., and Theis, F. J. (2022). Scarches: Preserving single-cell representations across multiple studies. *Nature Biotechnology*, 40:463–471.

Zhang, R., Wang, H., Li, X., and Wang, B. (2023). Scviva: Spatially aware variational inference for integrative single-cell analysis. *bioRxiv*.

Appendix-I: SCArches Implementation

- **Goal:** Map Xenium cells into a well-structured latent space learned from high-coverage scRNA-seq, without retraining the entire model.
- **Step 1: Train VAE on reference scRNA-seq**
  - Learns a latent manifold that captures biological variation.
  - Encoder produces $(\mu, \log \sigma^2)$ for latent variable z.
- **Step 2: Apply SCArches "architectural surgery"**
  - Freeze the *decoder* to preserve the reference manifold.
  - Fine-tune a new *query encoder* on Xenium while keeping the decoder fixed.
- **Step 3: Add alignment- and spatial-aware losses**
  - MMD loss for cross-modal distribution matching.
  - Spatial smoothness loss via kNN graph over Xenium coordinates.
  - Encoder proximity penalty to avoid over-shifting from the reference encoder.
- **Step 4: Train a lightweight classifier**
  - Train on reference latent embeddings.
  - Apply to Xenium embeddings for label transfer.