Cross-Modal Imputation and Uncertainty Estimation for Spatial Transcriptomics

Xiangyu Guo
Duke University

Ricardo Henao
Duke University

Abstract

High-resolution spatial transcriptomics (ST) technologies can capture gene expression at the cellular level along with spatial information, but are limited in the number of genes that can be profiled. In contrast, single-cell RNA sequencing (SC) provides more comprehensive gene expression profiles but lacks spatial context. To bridge these gaps, existing methods typically focus on single-modality prediction tasks, leveraging complementary information from the other modality. Here, we propose an attention-based cross-modal framework that simultaneously imputes gene expression for ST and recovers spatial locations for SC, while also providing uncertainty estimates for the expression of the imputed genes. Our approach was evaluated on three real-world datasets, where it consistently outperformed state-of-the-art methods in spatial gene profile imputation. Moreover, our framework enhances latent embedding integration between the two modalities, resulting in more accurate spatial position estimates.

1 INTRODUCTION

Understanding the spatial context of individual cells plays a vital role in deciphering their function in multicellular organisms (Marx, 2021). Single cell RNA-seq technology (scRNA-seq) has made it possible to characterize cell states and gene expression patterns (Liu and Trapnell, 2016). However, the spatial locations of each individual cell are lost while dissociating the tissues/organ and isolating cells. To this end, emerging spatially resolved transcriptomics (ST) technologies have been developed for transcriptomics profiling while retaining intact tissue structure and spatial information. Mainstream ST methods fall into two categories (Marx, 2021): image-based in-situ technologies such as sm-FISH, seq-FISH, and MERFISH, which usually have very high resolution but low throughput, so they can only detect a limited number of genes (Asp et al., 2020); while the other category utilizes spatial bar-coding and ex situ RNA-seq, such as 10X Visium, which can achieve relatively high throughput and capture thousands of genes in each spot but with low resolution and sensitivity (Asp et al., 2020).

Several methods have been proposed to impute high-resolution ST gene profiles by harnessing complementary gene expression from SC (Lopez et al., 2019; Abdelaal et al., 2020; Haviv et al., 2024). For example, gimVI jointly modeled scRNA-seq and ST data into a shared latent space with a variational autoencoder and encouraged the integration of SC and ST by adversarial training. Similarly, a more recent work proposed by Haviv et al. (2024) also jointly embedded SC and ST into a common latent space, but included two decoders to impute gene profiles and spatial covariance structures. Tangram (Biancalani et al., 2021) learned a matrix to map SC expression data to ST gene profiles while also providing cell-type deconvolution results for spatial spots. Instead, another direction of research focused on leveraging the relationship between gene expression and spatial location to recover spatial physical information in SC (Zhang et al., 2023; Qian et al., 2023; Li et al., 2024b,a).

Unlike previous work, the method we propose utilizes two encoders, one for SC and another one for ST, to capture information from both modalities separately. In order to improve cross-modal representation alignment, we leverage both a cross-attention module and contrastive learning, which also help to further improve spatial location recovery for SC. Our contributions can be summarized as follows.

- We propose an attention-based contrastive learning framework to cross-align ST and SC modalities in a unified shared latent space.

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

- We propose a sampling strategy to construct pairs of cells and spots to improve the efficiency and efficacy of the proposed model, which is trained with variational inference and contrastive learning.
- The proposed model produces uncertainty estimates for imputed genes that can be readily used to determine the reliability of their imputation. Moreover, the model can also be used to estimate the spatial location of SC data.
- Results on three real-world datasets demonstrate that the proposed approach outperforms competing state-of-the-art methods in terms of imputation, calibration, and proxies for spatial recovery.

2 RELATED WORK

Spatial gene profile imputation Spatial gene profile imputation can be categorized into methods that either impute missing values from the data by leveraging its internal covariance structure or information from other modalities, such as scRNA-seq or histological images. The former typically assumes that observed gene expression counts are influenced by sequencing noise, often modeled using Negative Binomial (NB) or Zero-Inflated Negative Binomial (ZINB) distributions, distinguishing true biological zeros (undetectable expression) from technical dropouts, as seen in methods such as DCA (Eraslan et al., 2019) and scVI (Lopez et al., 2018). More recently, transformer-based approaches like SpaFormer (Wen et al., 2024) have been developed to incorporate spatial information directly into the imputation process to better recover spatial gene expression pattern. Another line of work focuses on the integration of spatial transcriptomics data with histological images to impute gene expression in unmeasured areas, particularly in low-resolution ST datasets. These methods are less relevant to the method proposed here because we do not consider images; however, they are briefly addressed in the Appendix for completeness. Our work focuses on imputing gene expression profiles that are not measured in high-resolution ST with the auxiliary gene profiles from SC, along with the uncertainty estimates for the imputed genes. Like previously mentioned, several recent methods Haviv et al. (2024); Qiao and Huang (2024); Biancalani et al. (2021), have aimed to address the same problem using the shared observed genes between SC and ST, either by learning a shared latent space (ENVI) or by learning a mapping matrix (Tangram and TransImp) from cell to spot to impute spatial gene profile.

Uncertainty estimation Uncertainty estimates for imputation enable one to focus on more reliable genes so that helping us improve downstream analysis like spatial variable gene detection and spatial clustering. Two recent works have been proposed to address this problem. Sun et al. (2024) employed conformal inference to estimate the prediction interval for each gene in every spot and guided the downstream analysis by retrieving high-quality predictions. Another recent work presented TransImp (Qiao and Huang, 2024), which learned the mapping relationship from SC to ST on shared genes and obtained uncertainty measures for single genes by calculating the variance of multiple bootstrapping prediction results. gimVI (Lopez et al., 2019) generated the uncertainty measure for each gene by sampling multiple times from the variational posterior. Regarding our work, we are interested in quantifying the uncertainty of the dropout events (high sparsity) for each gene in ST, which allows for a more reliable spatial imputation of gene profiles.

Cross-modal alignment For applications in multi-omics integration in biology, deep generative modeling is usually employed to embed multi-modal data into a shared latent space (Ashuach et al., 2023; Cohen Kalafut et al., 2023; Cao et al., 2024). JAMIE takes partially matched single-cell multimodal data by variational autoencoder to perform cross-modal imputation. Similarly, MultiVI learned multimodal latent embedding separately and then fused them via a weighted nearest-neighbor scheme. Since JAMIE requires the model input to be partial matched single-cell multi-modal data and MultiVI considers other modalities such as ATAC-seq, which are not applicable in our scenario, we do not consider them in our experiments.

Spatial location recovery Spatial location recovery involves inferring the original spatial location of cells in SC. Several methods have been developed for this purpose. CeLEry (Zhang et al., 2023) utilized a deep neural network trained on ST along with elliptical quantile loss, which allowed location recovery in consecutive ST slices and uncertainty quantification. Other works sought to embed SC and ST into an unified space and then performed pseudo-space reconstruction based on the shared latent embedding (Qian et al., 2023; Li et al., 2024b), which enabled better generality and spatial location recovery accuracy for SC. However, instead of using cross-attention and contrastive learning, they use distribution matching approaches such as minimum mean discrepancy.

3 METHODS

3.1 Problem Definition

We consider two widely used sequencing technologies: scRNA-seq (SC) and spatial transcriptomics (ST). The

gene profile for a spot is defined as $x^s \in \mathcal{X}_s^{|G|}$, where G is the set of genes profiled by ST, and its corresponding spatial coordinates are denoted as $y^s \in \mathcal{Y}^2$. We use $\{s_n\}_{n=1}^{N_s}$ to denote the set of all spots in ST of size N_s, while $X^s \in \mathcal{X}_s^{|G|\times N_s}$ and $Y^s \in \mathcal{Y}^{2\times N_s}$ are the expression and spatial location matrices, respectively. Similarly, the gene profile for a cell is defined as $x^c \in \mathcal{X}_c^{|G|+|G'|}$, where we have assumed that the collection of genes profiled by SC, $G \cup G'$, is a super set of that of ST, G. Correspondingly, we use $\{c_n\}_{n=1}^{N_c}$ to denote the set of all cells in SC of size N_c and let $X^c \in \mathcal{X}_c^{(|G|+|G'|)\times N_c}$ be the SC expression matrix. Our objective is to utilize data from both sources to address the two main tasks, namely, i) imputing expression for genes not present in ST, *i.e.* G', by leveraging the more comprehensive gene profiles from SC, while providing uncertainty estimates for the imputed genes; and ii) spatial location recovery for cells from SC by utilizing the spatial information in ST.

The domains for SC and ST denoted as $\mathcal{X}_c$ and $\mathcal{X}_s$, respectively, are typically non-negative real numbers that accommodate for counts or fractional counts. Moreover, in the event that genes in ST are not a subset of those in SC, we simply select the set of shared observed genes between SC and ST.

3.2 Model Architecture

We address the aforementioned problem using a deep generative modeling approach illustrated in Figure 1. The model architecture consists of two encoder arms. The first encoder in the single cell (SC) branch denoted as $h^c = f_{\phi^c}(\cdot)$, and parameterized by ϕ^c, takes the cell gene profile x^c as input and produces cell embedding h^c. Similarly, the second encoder in the spatial transcriptomics (ST) branch, denoted as $h^s = f_{\phi^s}(\cdot)$, takes a spot gene profile x^s as input and outputs the spot embedding h^s. Notably, both encoders use only the shared set of genes G. This is done because the objective is to perform cross-modal representation alignment; thus, we would like gene profiles across modalities to be as comparable as possible. At the core of our architecture is the dual cross-attention (DCA) module, which aggregates the two embeddings h^c and h^s. The DCA module produces contextual source-specific representations by aligning a cell within the spatial context of spots or a spot within the cell context, depending on which modality serves as the query. This produces a latent space representation $z^{(i)} \in \mathbb{R}^d$ where $i \in \{c, s\}$, is the modality indicator and d is the dimension in the latent space. Moreover, the architecture includes two decoders: the first decoder $\hat{x}^{(i)} = f_\theta(z^{(i)})$, parameterized with θ, takes the modality-specific latent embedding $z^{(i)}$ as input and outputs the reconstructed gene expression profile for each modality $\hat{x}^{(i)}$, where $i = \{s, c\}$. The second decoder $\hat{y}^{(i)} = f_\eta(\cdot)$, is parameterized with η and is responsible for reconstructing the spatial positions $\hat{y}^{(i)}$. Since spatial information is only available for ST, the second decoder is trained only with the spot latent embeddings, z^s, and then used for inference with spatial location $\hat{y}^c$ for SC.

3.2.1 Dual Cross Attention Module

In order to effectively leverage the cross-modal information between SC and ST, we propose a dual cross-attention (DCA) module shown in Figure 1. The module consists of two parallel cross-attention modules (Vaswani et al., 2023). The first cross-attention uses the SC embedding h^c as the *query* and the ST embeddings h^s as *keys* and *values*. Its objective is to produce SC embeddings contextualized as weighted averages of ST embeddings (values), and whose weights are obtained via similarities between SC (query) and ST (key) embeddings. Similarly, the second cross-attention module contextualizes a ST embedding with SC embeddings. The attention component of the DCA module is formulated as follows.

$$Atte(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V\,, \qquad (1)$$

with $Q = f_{\phi^{(i)}}(x^{(i)})W_Q^{(i)}$, $K = f_{\phi^{(\setminus i)}}(x^{(\setminus i)})W_K^{(\setminus i)}$, $V = f_{\phi^{(\setminus i)}}(x^{(\setminus i)})W_V^{(\setminus i)}$, where $i \in \{c, s\}$ indicates the modality, SC and ST, respectively, and the notation $(\setminus i)$ indicates the modality counterpart to (i). The matrices $W_Q^{(i)} \in \mathbb{R}^{d^{(i)}\times d^{(i)}}$, $W_K^{(\setminus i)} \in \mathbb{R}^{d^{(\setminus i)}\times d^{(\setminus i)}}$, and $W_V^{(\setminus i)} \in \mathbb{R}^{d^{(\setminus i)}\times d^{(\setminus i)}}$ are learnable projections for queries, keys, and values, respectively. Here, $d^{(i)} = d^{(\setminus i)}$ denote the dimensions of the embeddings $h^{(i)}$. Like in the standard Transformer formulation, we can readily extend the above to multi-head attention. We complete the cross-attention module consistent with a Transformer layer, *i.e.*, the output of (1) is added to the query (via skip connection) and normalized before feeding it to a feedforward layer which is also subsequently normalized after the addition of a skip connection to produce the contextualized embeddings $(\mu_z^{(i)}, \sigma_z^{(i)}) = \text{DCA}(x^{(i)})$, where $\mu_z^{(i)}$ and $\sigma_z^{(i)}$ are the mean and variance of the latent embeddings for a spot or a cell using $(i) = \{c, s\}$, respectively. The use of this probabilistic embedding representation will become more apparent when we introduce the variational learning formulation in Section 3.3.1.

In Appendix C.1, we run an ablation study demonstrating the efficacy of the DCA module, compared to the model with standard self-attention applied independently to each modality when training the model.

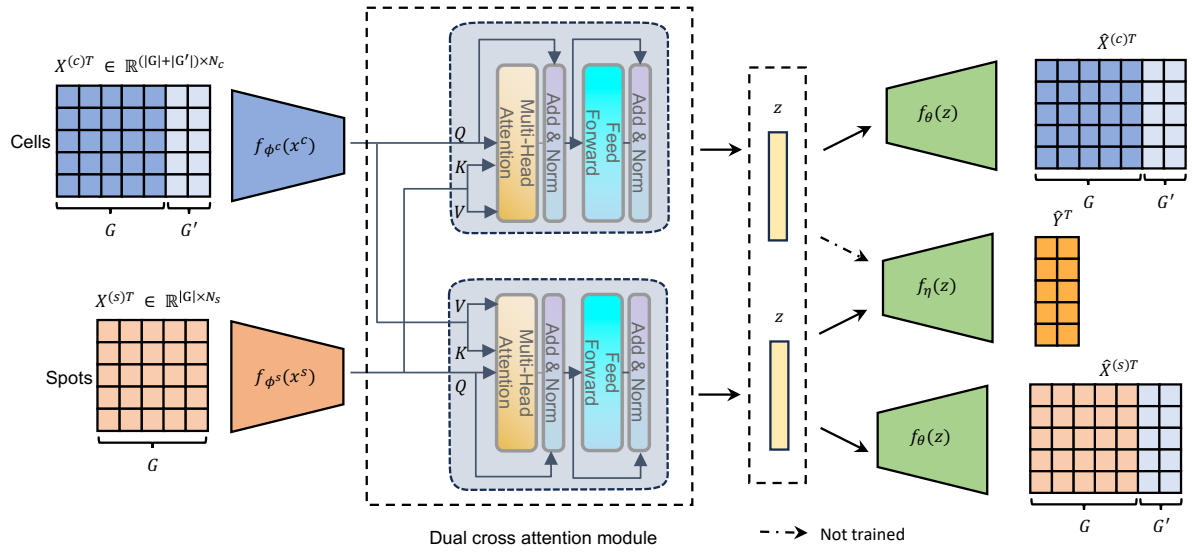

Figure 1: Proposed model architecture. Given the SC and ST data matrices $X^c \in \mathcal{X}_c^{(|G|+|G'|)\times N_c}$ and $X^s \in \mathcal{X}_s^{|G|\times N_s}$, respectively, two encoders (left) and the dual cross attention module (middle) produce the latent embedding z^i, then the two decoders (right) reconstruct the gene profile $\hat{X}^{(i)}$ and reconstruct the spatial location $\hat{Y}^{(i)}$ for both modalities, where $i \in \{c, s\}$ for CS and ST, respectively.

3.2.2 Cross-modal and Contrastive Pairs

In practice, we train models using stochastic learning with minibatches, thus assume that we have a randomly sampled subset of cells $\{c_n\}_{n=1}^b$ from $\{c_n\}_{n=1}^{N_c}$ and a randomly sampled set of spots $\{s_n\}_{n=1}^b$ from $\{s_n\}_{n=1}^{N_s}$ as input to the model, where b is the minibatch size. Computing cross attention between the embeddings of randomly sampled cells and spots embedding is highly inefficient, since instead, we would rather have similar sets of spots and cells so their attention weights in (1) are not too close to zero, thus effectively contributing to the weighted average. However, due to the typical size of a cell population (N_c is usually in the 10k to 100k range) and the cellular heterogeneity of the sample (tissue) used for ST, the chances to select similar cells and spots at random are fairly low, which will not prevent the model from learning, but it will certainly do so inefficiently. So motivated, we would like to form SC-ST pairs that fulfill the following two requirements: *i*) finding spots to be paired by similarity to randomly sampled cells to improve the efficacy of the cross-attention module, and *ii*) finding cells and spots whose latent representations z^c and z^s, respectively, are likely to be close in latent space. The latter will be useful for contrastive learning, which will be described in Section 3.3.2.

To effectively form SC-ST pairs for cross-modal attention computation, we first compute the full pairwise cosine similarity matrix between all cells and spots: $M = \frac{x_i^c \cdot x_j^s}{\|x_i^c\| \|x_j^s\|}$, for $i = 1, \ldots, N_c$ and $j = 1, \ldots, N_s$. As shown in Figure 2, the similarity submatrix for sampled cells is denoted as $M^c \in \mathbb{R}^{b\times N_s}$. We can approximate the similarity for all spots relative to the mini batch of b randomly selected cells as $P_s = \text{softmax}\left(\mathbf{1}_b^T M^c\right)$, where $\mathbf{1}_b$ is a vector of ones of length b. The aggregated similarity P_s represents the normalized average similarity of a given spot wrt the cells in the set $\{c_n\}_{n=1}^b$. Subsequently, the pairwise cross-modal spots $\{s_n\}_{n=1}^b$ with batch size b are sampled according to P_s, which can be interpreted as a distribution over all spots, $\{s_n\}_{n=1}^{N_s}$.

Moreover, we also seek to generate a latent representation in which contextualized embeddings z^s and z^c for SC and ST, respectively, are likely to be aligned. This can be done via contrastive learning, however, to do so we require *positive pairs* for a minibatch of cells $\{c_n\}_{n=1}^b$. Instead of naively randomly sampling a minibatch of spots $\{s_n\}_{n=1}^b$, and then defining which of the cell-spot pairs can be used as positive pairs, we leverage the cosine similarity matrix M from above. Specifically, we first rank the cosine similarities for a given cell c_n in $\{c_n\}_{n=1}^b$ wrt all spots in $\{s_n\}_{n=1}^{N_s}$, *i.e.*, $r_n = \text{rank}(m_n)$, where r_n is a N_s-dimensional vector containing the ranks of the similarity scores for cell c_n

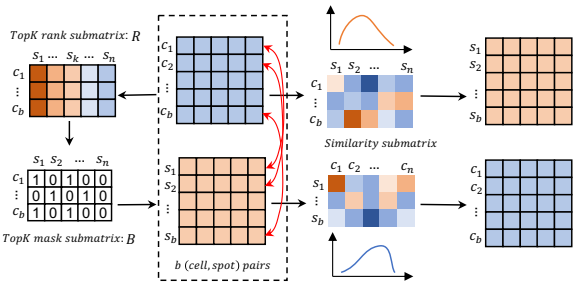

Figure 2: Cross modal and contrastive pair sampling. Given a randomly sampled subset of cells $\{c_n\}_{n=1}^b$ (the blue matrix in the dashed line), we get the average similarity P_s from the similarity sub-matrix M^c (middle right), and the sampled pairwise cross-modal spots $\{s_n\}_{n=1}^b$ (upper right). Top and bottom left show the rank matrix R for $\{c_n\}_{n=1}^b$ and the binary indicator matrix: B, respectively, which produces the sampled top K spots matrix (yellow matrix in the dashed line), the marginal distribution P_c and the subset of cells $\{c_n\}_{n=1}^b$ sampled from it.

in descending order and obtained for row m_n of M. Further, R is a matrix composed of $\{r_n\}_{n=1}^b$ rows with the ranks for the complete mini batch $\{c_n\}_{n=1}^b$. Then, we can use R to select the top K spots for each cell, which can be done effectively by thresholding R to obtain a matrix B of the same size, but with elements $b_{ij} = \mathbb{I}[r_{ij} \leq K]$, where $\mathbb{I}[\cdot]$ is the indicator function, thus $b_{ij} = 1$, if spot s_j is ranked among the top K for cell c_i, and $b_{ij} = 0$ otherwise. We set $K = 50$ in our experiments. This pairing scheme enables contrastive learning to pull similar cells and spots (with the same cell type) closer in the latent space while pushing dissimilar pairs apart.

In Appendix C.2 we show an ablation study demonstrating the efficacy of the cross-modal and contrastive pair construction summarized in Figure 2, compared to sampling cells and spots at random from the full dataset when training the model.

3.3 Learning Objective

We optimize our model using three terms, namely, a reconstruction error for each modality, a regularization term for the latent space, and a contrastive loss to align the latent representation between SC and ST. The former two loss terms are from the standard variational autoencoder (VAE) (Kingma and Welling, 2022).

3.3.1 Variational Inference

For either a cell or spot indexed by n, let the prior distribution for the latent variable be $z_n^{(i)} \sim \mathcal{N}(0, I_d)$. For convenience, we let $z_n^{(i)}$ follow a multivariate Gaussian distribution, with zero mean and unit spherical variance, where I_d is the identity matrix of size d.

For the ST data which manifest as counts, we assume a negative binomial (NB) likelihood. This is justified by high-resolution hybridization data, which in general have high sensitivity, and existing work (Zhao et al., 2022) demonstrating that the negative binomial is sufficient for modeling the majority of genes across ST technologies. Formally, we write

$$p(x_{ng}^s | l_n^s, \mu_{ng}^s, \theta_g^s) = \text{NB}(l_n^s \mu_{ng}^s, \theta_g^s)\,, \tag{2}$$

where x_{ng}^s denotes the a gene expression for spot n and gene $g \in G$, and l_n^s, μ_{ng}^s and θ_g^s are the library size, normalized gene expression frequency and inverse dispersion for gene g in spot n, respectively. Using these parameters, we can estimate the expected gene expression for the gene g in spot n as $\mathbb{E}[x_{ng}^s] = l_n^s \mu_{ng}^s$.

For SC data, we instead assume zero-inflated negative binomial (ZINB) likelihood to accommodate for excess sparsity in the data, *i.e.*, we typically expect more cells with zero gene expression since scRNA-seq data relatively has more technical noise and lower sequencing depth compared to high resolution hybridization ST data (Moses and Pachter, 2022). We write

$$p(x_{ng}^c | l_n^c, \mu_{ng}^c, \theta_g^c, \pi_{ng}^c) = \text{ZINB}(l_n^c \mu_{ng}^c, \theta_g^c, \pi_{ng}^c)\,, \tag{3}$$

where x_{ng}^c denotes the a gene expression for cell n and $g \in G \cup G'$, and l_n^c, μ_{ng}^c, θ_g^c and π_{ng}^c are the library size, normalized gene expression frequency, inverse dispersion, and the dropout rate for the gene g in spot n, respectively. Using these parameters, we can estimate the expected gene expression for the gene g in spot n as $\mathbb{E}[x_{ng}^c] = (1 - \pi_{ng}^c) l_n^c \mu_{ng}^c$.

The parameters of the likelihood for ST and SC data in (2) and (3), respectively, are modeled as feedforward neural networks. Specifically, we let $(\mu_{ng}^{(i)}, \theta_g^{(i)}, \pi_{ng}^{(i)}) = f_\theta(z^{(i)})$, where $f_\theta(\cdot)$ is the shared decoder introduced in Section 3.2. For the library size, we estimate $l_n^{(i)}$ directly from the data, but separately for each modality via $\mathbf{1}^T \log(X^{(i)})$, where $\mathbf{1}$ is a vector of ones of size $(|G| + |G'|)$ or $|G|$, depending on the modality. In Appendix A.2, we show how to learn $l_n^{(i)}$ from data using the variational inference formalism by setting a log-normal distribution as prior.

Since the marginal likelihood $p(x^{(i)})$ for the likelihoods in (2) and (3) are often intractable to compute directly, we introduce a variational approximation to the true posterior $p(z^{(i)}|x^{(i)})$, and derive a evidence lower bound (ELBO) (Kingma and Welling, 2022) as

$$\log p_\theta(x^{(i)}) \geq \mathbb{E}_{q_\psi(z^{(i)}|x^{(i)})} \left[\log \frac{p_\theta(x^{(i)}, z^{(i)})}{q_\psi(z^{(i)}|x^{(i)})} \right], \tag{4}$$

where $q_\psi(z^{(i)}|\mu_z^{(i)}, \sigma_z^{(i)})$ is obtained from the cross-attention modules for each modality via $(\mu_z^{(i)}, \sigma_z^{(i)}) = \mathrm{DCA}(x^{(i)})$, and the joint distribution $p_\theta(x^{(i)}, z^{(i)})$ is obtained for each modality via the Gaussian prior and likelihoods in (2) and (3) for ST and SC, respectively. Specifically, we would like to estimate parameters ψ and θ to maximize the ELBO for each modality, which is equivalent to minimize the loss function below.

$$\mathcal{L}_{\mathrm{VAE}}^{(i)} = \underbrace{-\mathbb{E}_{q_\psi(z^{(i)}|x^{(i)})}[\log p_\theta(x^{(i)}|z^{(i)})]}_{\text{Negative log likelihood}} + \underbrace{\mathrm{KL}(q_\psi(z^{(i)}|x^{(i)})\|p(z^{(i)})\}}_{\text{Regularization term}} \tag{5}$$

where the first term denotes modality-specific negative log likelihood for ST and SC in (3) and (2), respectively. The regularization term is the Kullback–Leibler (KL) divergence, which encourages the distribution of the variational posteriors $q_\psi(z^{(i)}|x^{(i)})$ to be as close as possible to $p(z^{(i)})$ (a standard Gaussian).

3.3.2 Contrastive Cross-Modal Alignment

Recall that in order to encourage the alignment of embeddings across modalities, we proposed a sampling strategy to form SC-ST pairs for contrastive learning, and we retrieve a collection of spots $\{s_n\}_{n=1}^b$ such that for every single cell c_i in $\{c_n\}_{n=1}^b$, there exists at least one positive pair between cells and spots. If we let S_b the set of all positive pairs in $\{s_n\}_{n=1}^b$ and $\{c_n\}_{n=1}^b$ constructed as described in Section 3.2.2. We can use the following contrastive loss (Chen et al., 2020)

$$\mathcal{L}_{\mathrm{CL}} = -A \sum_{(i,j)\in S_b} \log \frac{\exp(u_{ij}/\tau)}{B\sum_{(i',j')\in S_b'} \exp(u_{i'j'}/\tau)}, \tag{6}$$

where $u_{ij} = \mathrm{sim}(z_i^c, z_j^s)$ is the cosine similarity between the embeddings for spot z_i^s and cell z_i^c and in a slight abuse of notation, S_b' is a set containing all pairs in mini batch $\{s_n\}_{n=1}^b$ and $\{c_n\}_{n=1}^b$ that are not positive pairs, *i.e.*, the complement of S_b. Further, $A = 1/|C_b|$, $B = 1/|S_b'|$, and τ is a temperature parameter that controls the sharpness of the cosine similarity. We included different temperature parameters τ during contrastive learning in Appendix C.5 to investigate its impact on final model performance.

3.3.3 Full Reconstruction Loss

The overall loss for cross-modal training is

$$\mathcal{L} = \mathcal{L}_{\mathrm{VAE}}^{(c)} + \mathcal{L}_{\mathrm{VAE}}^{(s)} + \lambda\mathcal{L}_{\mathrm{CL}}, \tag{7}$$

where $\mathcal{L}_{\mathrm{VAE}}^{(i)}$ and $\mathcal{L}_{\mathrm{CL}}$ are in (8) and (6), respectively, λ is a hyperparameter that balances contrastive cross-modal fusion and single-modality statistics, and we set $\lambda = 5$ in our experiments.

3.3.4 Supervised Spatial Location Training

Initially we intended to encode the spatial coordinates for ST by adding positional encoding to the ST encoder branch and then decode the position information. However, it is difficult to coordinate the training losses between spatial location recovery and spatial gene imputation, as it often perturbs the integration of the two modalities in latent space. Consequently, in our current implementation, we separate the problem into two distinct training stages. Firstly, we train the deep-generative model to obtain the representation for cells and spots $z^{(i)}$. Then we train a decoder only for ST latent embedding and its spatial location. Specifically, we write $\hat{y}^{(i)} = f_\eta(z^{(i)})$, where $i \in \{c, s\}$, and we train a multilayer perceptron with ST data representation and its physical location. Then the pseudo-location for SC, $\hat{y}^{(c)}$, can be inferred with the trained model. In Appendix C.3 we show experiments the effect of training $f_\eta(z^{(i)})$ separately as described here and jointly with the full reconstruction loss in (7).

4 EXPERIMENTS

We begin comparing our model to other state-of-the-art methods, including gimVI, Tangram, SpaGE and ENVI, for ST gene profile imputation across three datasets. Next, we present calibration results for the imputed genes. Further, we evaluate the cross-modal alignment of the latent embeddings produced by our model against those generated by gimVI and ENVI. Finally, we show the performance of our model in recovering spatial locations, comparing it to CeLEry and scSpace.

Datasets We consider three real-world single-cell (SC) and spatial transcriptomics (ST) datasets to validate and benchmark our model. The first dataset consists of (seqFISH) ST data on mouse organogenesis and SC data from a mouse embryo at the E8.5 stage. The second dataset includes (MERFISH) ST data from a spatial cell atlas of the whole mouse brain, along with SC data for the cortical and hippocampus regions. The third dataset comprises (MERFISH) ST and (Smart-seq) SC data from the mouse primary visual cortex (VISP). Links and a dataset summaries are provided in Appendix B.1. These datasets cover a variety of biological contexts, with ST data ranging from 2,376 to 19,416 spots and SC data ranging from 4,633 to 14,249 cells. Shared genes across the datasets range from 230 to 320.

Baselines We compare our model with a naive baseline, gimVI, Tangram, SpaGE and ENVI on spatial gene imputation, and with CeLEry and scSpace on

	Mouse embryo				Mouse brain				Mouse VISP			
	Pearson	Spearman	Kendall tau	RMSE	Pearson	Spearman	Kendall tau	RMSE	Pearson	Spearman	Kendall tau	RMSE
Naive	0.395	**0.347**	**0.276**	9.433†	0.302	0.234	0.187	2.282†	0.354†	0.319	0.252	247.841†
SpaGE	0.255†	0.179†	0.145†	1.181	NA	NA	NA	NA	0.238†	0.184†	0.146†	8.186
gimVI	0.379†	0.319†	0.254†	1.255	0.284	0.233†	0.185†	1.745	0.334†	0.312†	0.246†	**7.963**
Tangram	0.366†	0.288†	0.228†	1.194	0.261†	0.193†	0.153†	1.620	0.342†	0.272†	0.213†	12.896†
ENVI	0.391	0.325	0.258	1.323†	**0.336**	0.251	0.201	1.943	0.102†	0.060†	0.045†	76.786†
Ours	**0.405**	0.336	0.267	**1.169**	0.315	**0.261**	**0.209**	**1.690**	**0.386**	**0.345**	**0.273**	8.029

Table 1: Spatial gene imputation performance on three datasets. † indicates our results are significantly better ($p < 0.05$) than the corresponding method. The best results are highlighted in **bold**.

spatial location recovery performance. The naive baseline is calculated from the cosine similarity between cells and spot, which uses the top K similarity cells to impute the unmeasured gene profile in a given spot. Additional details including training, parameter settings, and running time for each method are provided in Appendix B.2.

Architecture and training details The two encoders consist of a one-layer MLP with hidden dimension 64, which projects high-dimensional gene profile features in a low dimension. They are followed by multi-head cross attention with four heads and feed-forward layer with 64 neurons. The latent dimension is set to 20. The decoder for gene profiles is set to three fully connected layers, each decoding the parameters for the ZINB or NB distribution. We set the top $K = 50$ and train the model for 200 epochs with the Adam optimizer and learning rate 0.001. For the spatial location recovery, the model is a MLP with layers (20, 64 and 32 units) producing spatial coordinate outputs. This is optimized with the Adam optimizer to minimize the $L1$ loss between the true spatial position y^s and the predicted spatial location $\hat{y}^s$. All the experiments are run on a workstation with 128G RAM, 48 core CPU, single NVIDIA RTX 4090 GPU. The learning rate is set to 0.01 and we train for 100 epochs.

To validate our model's imputation performance, we randomly held out a fraction of genes from the ST data for evaluation. Specifically, we conducted 5-fold cross-validation across all three datasets, ensuring that every gene in ST was systematically evaluated across all spatial spots. To maintain fairness in comparison with other methods, we used a consistent random split seed across experiments. The source code used to run the experiments and the implementation details are publicly available on GitHub[1].

Evaluation metrics To evaluate the spatial gene profile imputation performance, we employ Pearson, Spearman's rank, and Kendall tau correlations. Spearman is a more reliable metrics compared to Pearson, especially when the correlation between the imputed gene profile and ground-truth gene expression is not linear. Kendall tau is more suitable for comparisons involving repeating (tie) values due to excessive zero expression values. Using the likelihood definitions in (3) and (2), we can estimate the dropout probability (see Appendix A.1 for details), and use the AUC (ROC) score to estimate the model performance on zero-count prediction for all genes in spots. Further calibration plots are produced and summarized via ordinary least squares fit in terms of calibration slope and intercept (details in Appendix B.3). The integration of SC and ST modality is evaluated by k-nearest neighbor's purity and the batch average silhouette width (bASW). For the former, we focus on local cell-type similarity, while for the latter we estimate (globally) whether cell-types for the two modalities are well clustered in latent space. The accuracy of spatial location recovery for SC is calculated as k-nearest neighbor's purity belonging to the same cell type in ST since we do not have ground truth for SC. Accordingly, we use the Mann-Whitney U test to test the significant difference between our model's results and other methods among the aforementioned metrics. additional details about the statistical test are provided in Appendix B.4.

4.1 Results

Spatial gene profile imputation We benchmark our model imputation results with related methods, including gimVI, Tangram, SpaGE, and ENVI on mouse embryo seqFISH data, mouse brain MERFISH data, and mouse VISP data. Results in Table 1 and in Appendix C.8 show that our model consistently outperformed gimVI, SpaGE and Tangram, ENVI across the three datasets. Although ENVI is slightly better than our method in terms of Pearson's correlation on the mouse brain dataset, our method outperforms ENVI in the other four metrics. Notably, ENVI failed to achieve optimal performance on the mouse VISP dataset because of the high sparsity in SC's gene expression profile. The average sparsity for genes in the VISP SC data is (89.3%). Although our method is more robust and significantly better than other methods, it is interesting to see that neither our method or

[1]https://github.com/woweizhi/Cross-modal-Imputation-Uncertainty-Estimation-ST.

Dataset	Method	AUC	Calib Slope	Calib Intercept
MouseEmbryo	gimVI	0.697†	**0.998**	**-0.107**
	our	**0.710**	1.286	-0.299
MouseBrain	gimVI	0.698†	0.652†	0.293†
	our	**0.722**	**1.033**	**0.019**
MouseVISP	gimVI	0.715†	0.717†	0.245†
	our	**0.738**	**0.836**	**0.195**

Table 2: Calibration results on zero count prediction between gimVI and our model. † indicates our results are significantly better ($p < 0.05$) than gimVI. The best results are in **bold**.

Dataset	Method	knn5	knn10	knn20	knn50	bASW
MouseEmbryo	gimVI	**0.847**	**0.822**	**0.805**	**0.785**	0.770
	ENVI	0.777†	0.735†	0.702†	0.663†	0.753
	our	**0.847**	**0.822**	0.804	0.783	**0.807**
MouseBrain	gimVI	0.969†	0.961†	0.954†	0.944†	**0.836**
	ENVI	0.943†	0.919†	0.893†	0.851†	0.746
	our	**0.987**	**0.984**	**0.981**	**0.976**	0.806

Table 3: Multi-modal integration results based on kNN purity and average batch silhouette score (bASW). † indicates our results are significantly better ($p < 0.05$) than gimVI. The best results are in **bold**.

ENVI significantly outperformed the naive baseline in terms of correlation metrics. Notably, our approach is significantly better than the naive baseline in terms of RMSE, which, being based on cosine similarity, is unable to appropriately estimate the scale of gene expression profiles.

Spatial gene uncertainty estimates For each gene imputed in ST, the dropout rate (the probability of the gene count being zero) is estimated via (2). The results in Table 2 demonstrate that our model consistently outperforms gimVI in predicting zero counts across all three datasets (Mouse Embryo, Mouse Brain, and Mouse VISP), offering better AUC scores and improved calibration. Since only gimVI allow us to compute the zero probability (among all baseline methods), here we only compare our model with gimVI. While gimVI occasionally performs better in calibration slope for Mouse Embryo, the overall performance of our model, especially in minimizing bias (as indicated by calibration intercept), shows that ours is more robust and generalizable for spatial transcriptomics imputation tasks. These findings underscore the value of our approach in accurately predicting gene expression patterns while maintaining reliable probability estimates.

Cross-modal representation alignment We evaluated the performance of our method's cross-modal latent integration with gimVI and ENVI latent embedding in Mouse Embryo and Mouse Brain datasets, since they offered cell-type labels for every single cell/spot. Integration is evaluated based on k-nearest neighbors (kNN) purity; where purity is calculated for a given cell/spot as the proportion of k-nearest neighbors having the same cell type. Further, the batch-adjusted silhouette width (bASW) is used to quantify how well the two modalities (SC and ST) are integrated in latent space. As shown in Table 3, our method consistently achieves superior performance in maintaining kNN purity across both datasets. Specifically, in the Mouse Brain dataset, it achieves the highest kNN purity for every neighborhood size (k=5, 10, 20, 50), significantly outperforming both gimVI and ENVI. In the Mouse Embryo dataset, our method performs comparably to gimVI for smaller neighborhood sizes (k=5 and k=10), though gimVI slightly edges out in larger neighborhoods (k=20 and k=50). Nevertheless, our method remains highly competitive and provides strong local structure preservation, where it tends to group highly similar cell population together in latent space because of our sampling strategy. ENVI lags behind both models in kNN purity across all metrics, particularly in larger neighborhoods, where it shows weaker preservation of cell-type structure. Regarding the bASW metric, our method achieves the highest bASW score (0.807) in the Mouse Embryo dataset, highlighting its superior ability to integrate scRNA-seq and spatial transcriptomics while maintaining cell type consistency. In the Mouse Brain dataset, while gimVI slightly outperforms with a higher bASW score (0.836), our method still achieves a competitive score of 0.806, indicating a robust integration of the two modalities. Since gimVI is optimized using adversarial training, its objective is to integrate the two modalities in the latent space, which tends to favor the bASW metric as it assesses global structure. In contrast, our method focuses on identifying locally similar cell populations, leading to better performance in kNN but not in bASW.

Spatial location recovery To quantitatively evaluate our method's capability in spatial location recovery, we compared our model with two recently developed methods (CeLEry and scSpace), which are specially designed for the spatial location recovery task. Since we do not have ground truth spatial location for SC, we used the spatial position in ST as a reference, and the kNN purity is calculated as the percentage of cells whose k-nearest neighbors (in the latent space) belong to the same cell type in ST. As Table 4 shows, Our method consistently outperforms both CeLEry and scSpace across all k values in both Mouse Embryo and Mouse Brain datasets, indicating its advantage in recovering spatial locations while maintaining cell-type structure in the latent space. It is also notable that our method is robust across different neighbor-

Dataset	Method	knn5	knn10	knn20	knn50
MouseEmbryo	CeLEry	0.167†	0.169†	0.172†	0.176†
	scSpace	0.176†	0.178†	0.181†	0.187†
	our	**0.290**	**0.291**	**0.293**	**0.291**
MouseBrain	CeLEry	NA	NA	NA	NA
	scSpace	0.402†	0.406†	0.405†	0.395†
	our	**0.650**	**0.657**	**0.658**	**0.634**

Table 4: Spatial location recovery results for SC based on kNN purity in ST. † indicates our results are significantly better ($p < 0.05$) than gimVI. The best results are in **bold**.

hood sizes ($k = 5$, $k = 10$, $k = 20$, $k = 50$), while CeLEry and scSpace either show decreasing performance or show only slight improvements as the neighborhood size increases. Our method still achieves high kNN purity among various neighborhood sizes, demonstrating that it scales well across different levels of local neighborhood density.

In addition, we demonstrate in Figure 3 that our model is able to infer the relative precise spatial location for sub-cell types (CA3-do, CA3-ve, L6b ENT) that only exit in SC through the cross-modal embedding space. We later confirmed our inferred pseudo-space accuracy for sub-cell types from the relevant literature (Yao et al., 2021) and the anatomical structure of the mouse brain (Allen Institute for Brain Science, 2004). For example, the sub-cell type of CA3 exhibited spatial gradient from the ventral axis (CA3-do) to the dorsal axis (CA3-de) and the L6b ENT cell-type located in the medial temporal lobe, between the hippocampus and the neocortex.

5 CONCLUSION

We introduced an attention-based contrastive learning framework to cross-align SC and ST embeddings into a shared latent space via variational inference. We also proposed a cross-modal and contrastive pairs sampling strategy that effectively forms similar SC-ST pairs. Compared to other state-of-the-art methods, our approach accurately imputes gene profiles for ST along with uncertainty estimates, and provides precise spatial location recovery for SC.

6 LIMITATIONS

A primary limitation of our method, also seen in related approaches, is that our encoder only considers genes shared between SC and ST datasets, potentially narrowing the integrated representation. We also did not incorporate spatial position embeddings, missing an opportunity to capture tissue-specific context. Moreover, computing a cosine similarity matrix between all cells and spots can be computationally expensive for large datasets.

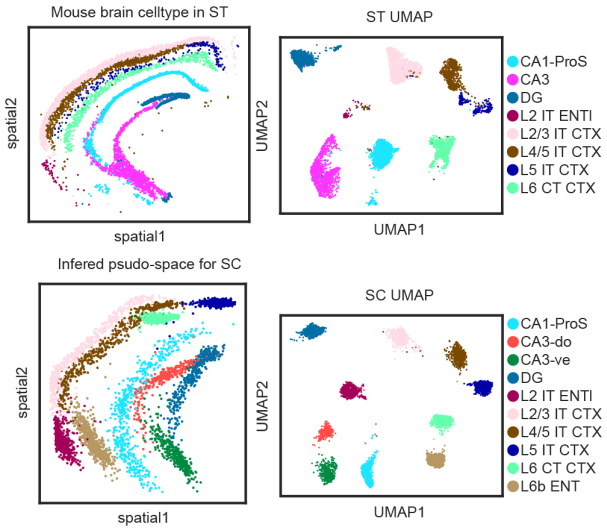

Figure 3: Top left: original spatial map of ST colored by cell types. Lower left: inferred spatial positions for SC. Top and Lower right: latent spaces of ST and SC visualized via UMAP, respectively.

To address these limitations, future work could enhance the spatial encoder with additional spatial features, *e.g.*, spot adjacency or tissue structure, to further strengthen cross-modal alignment. Adopting foundation models for scRNA-seq may also provide a more comprehensive gene profile characterization across cells and spots. Lastly, we plan to explore approximate nearest neighbor methods to efficiently scale to large datasets, reducing computational overhead.

References

Abdelaal, T., Mourragui, S., Mahfouz, A., and Reinders, M. J. T. (2020). Spage: Spatial gene enhancement using scrna-seq. *Nucleic Acids Research*, 48(18):e107–e107.

Allen Institute for Brain Science (2004). Allen mouse brain atlas. `https://mouse.brain-map.org/`. Accessed: 2024-10-10.

Ashuach, T., Gabitto, M. I., Koodli, R. V., Saldi, G.-A., Jordan, M. I., and Yosef, N. (2023). Multivi: deep generative model for the integration of multimodal data. *Nature Methods*, 20(8):1222–1231.

Asp, M., Bergenstråhle, J., and Lundeberg, J. (2020). Spatially resolved transcriptomes—next generation tools for tissue exploration. *BioEssays*, 42(10).

Biancalani, T., Scalia, G., Buffoni, L., Avasthi, R., Lu, Z., Sanger, A., Tokcan, N., Vanderburg, C. R., Segerstolpe, , Zhang, M., Avraham-Davidi, I., Vick-

ovic, S., Nitzan, M., Ma, S., Subramanian, A., Lipinski, M., Buenrostro, J., Brown, N. B., Fanelli, D., Zhuang, X., Macosko, E. Z., and Regev, A. (2021). Deep learning and alignment of spatially resolved single-cell transcriptomes with tangram. *Nature Methods*, 18(11):1352–1362.

Cao, Y., Zhao, X., Tang, S., Jiang, Q., Li, S., Li, S., and Chen, S. (2024). scbutterfly: a versatile single-cell cross-modality translation method via dual-aligned variational autoencoders. *Nature Communications*, 15(1).

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Cohen Kalafut, N., Huang, X., and Wang, D. (2023). Joint variational autoencoders for multimodal imputation and embedding. *Nature Machine Intelligence*, 5(6):631–642.

Eraslan, G., Simon, L. M., Mircea, M., Mueller, N. S., and Theis, F. J. (2019). Single-cell rna-seq denoising using a deep count autoencoder. *Nature Communications*, 10(1).

Haviv, D., Remšík, J., Gatie, M., Snopkowski, C., Takizawa, M., Pereira, N., Bashkin, J., Jovanovich, S., Nawy, T., Chaligne, R., Boire, A., Hadjantonakis, A.-K., and Pe'er, D. (2024). The covariance environment defines cellular niches for spatial inference. *Nature Biotechnology*.

Kingma, D. P. and Welling, M. (2022). Auto-encoding variational bayes.

Li, S., Ma, J., Zhao, T., Jia, Y., Liu, B., Luo, R., and Huang, Y. (2024a). Cellcontrast: Reconstructing spatial relationships in single-cell rna sequencing data via deep contrastive learning. *Patterns*, 5(8):101022.

Li, S., Shen, Q., and Zhang, S. (2024b). Spatial transcriptomics-aided localization for single-cell transcriptomics with stalocator.

Liu, S. and Trapnell, C. (2016). Single-cell transcriptome sequencing: recent advances and remaining challenges. *F1000Research*, 5:182.

Lopez, R., Nazaret, A., Langevin, M., Samaran, J., Regier, J., Jordan, M. I., and Yosef, N. (2019). A joint model of unpaired data from scrna-seq and spatial transcriptomics for imputing missing gene expression measurements.

Lopez, R., Regier, J., Cole, M. B., Jordan, M. I., and Yosef, N. (2018). Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 15(12):1053–1058.

Marx, V. (2021). Method of the year: spatially resolved transcriptomics. *Nature Methods*, 18(1):9–14.

Moses, L. and Pachter, L. (2022). Museum of spatial transcriptomics. *Nature Methods*, 19(5):534–546.

Qian, J., Liao, J., Liu, Z., Chi, Y., Fang, Y., Zheng, Y., Shao, X., Liu, B., Cui, Y., Guo, W., Hu, Y., Bao, H., Yang, P., Chen, Q., Li, M., Zhang, B., and Fan, X. (2023). Reconstruction of the cell pseudo-space from single-cell rna sequencing data with scspace. *Nature Communications*, 14(1).

Qiao, C. and Huang, Y. (2024). Reliable imputation of spatial transcriptomes with uncertainty estimation and spatial regularization. *Patterns*, 5(8):101021.

Sun, E. D., Ma, R., Navarro Negredo, P., Brunet, A., and Zou, J. (2024). Tissue: uncertainty-calibrated prediction of single-cell spatial transcriptomics improves downstream analyses. *Nature Methods*, 21(3):444–454.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2023). Attention is all you need.

Wen, H., Tang, W., Jin, W., Ding, J., Liu, R., Dai, X., Shi, F., Shang, L., Liu, H., and Xie, Y. (2024). Single cells are spatial tokens: Transformers for spatial transcriptomic data imputation.

Yao, Z., van Velthoven, C. T., Nguyen, T. N., Goldy, J., Sedeno-Cortes, A. E., Baftizadeh, F., Bertagnolli, D., Casper, T., Chiang, M., Crichton, K., Ding, S.-L., Fong, O., Garren, E., Glandon, A., Gouwens, N. W., Gray, J., Graybuck, L. T., Hawrylycz, M. J., Hirschstein, D., Kroll, M., Lathia, K., Lee, C., Levi, B., McMillen, D., Mok, S., Pham, T., Ren, Q., Rimorin, C., Shapovalova, N., Sulc, J., Sunkin, S. M., Tieu, M., Torkelson, A., Tung, H., Ward, K., Dee, N., Smith, K. A., Tasic, B., and Zeng, H. (2021). A taxonomy of transcriptomic cell types across the isocortex and hippocampal formation. *Cell*, 184(12):3222–3241.e26.

Zhang, Q., Jiang, S., Schroeder, A., Hu, J., Li, K., Zhang, B., Dai, D., Lee, E. B., Xiao, R., and Li, M. (2023). Leveraging spatial transcriptomics data to recover cell locations in single-cell rna-seq with celery. *Nature Communications*, 14(1).

Zhao, P., Zhu, J., Ma, Y., and Zhou, X. (2022). Modeling zero inflation is not necessary for spatial transcriptomics. *Genome Biology*, 23(1).

Checklist

1. For all models and algorithms presented, check if you include:

(a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
(b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [No]
(c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Not Applicable]
   (c) Clear explanations of any assumptions. [Not Applicable]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Appendix: Cross-Modal Imputation and Uncertainty Estimation for Spatial Transcriptomics

A METHODS

A.1 The Probability of Zero in Negative Binomial Distribution

Given a single gene g in spot n, by referring to 2, we could calculate the probability when count is zero:

$$p(x_{ng}^s = 0) = (\frac{\theta_g^s}{\theta_g^s + l_n^s \mu_{ng}^s})^{\theta_g^s}.$$

A.2 Learning Library Size from Data Using VAE

Instead empirically estimating the library size $l_n^{(i)}$ from the data, consistent with previous work by (Lopez et al., 2018), we assume the prior for library size for a spot or cell follows a log normal distribution. Formally,

$$p(l_n^{(i)}|l_\mu, l_\sigma^2) = \text{LogNormal}(l_\mu, l_\sigma^2),$$

where $i \in \{c, s\}$, and l_μ and l_σ^2 specify their mean and variance.

We then rewrite the ELBO as follows:

$$\log p_\theta(x^{(i)}) \geq \mathbb{E}_{q_\psi(z^{(i)}|x^{(i)})q_\psi(l^{(i)}|x^{(i)})}\left[\log \frac{p_\theta(x^{(i)}, z^{(i)}, l^{(i)})}{q_\psi(z^{(i)}|x^{(i)})q_\psi(l^{(i)}|x^{(i)})}\right],$$

where $q_\psi(z^{(i)}|\mu_z^{(i)}, \sigma_z^{(i)})$ is obtained from the cross attention modules for each modality via $(\mu_z^{(i)}, \sigma_z^{(i)}) = \text{DCA}(x^{(i)})$, and $q_\psi(l^{(i)}|\mu_l^{(i)}, \sigma_l^{(i)})$ simply obtained from a one-layer MLP for each modality via $(\mu_l^{(i)}, \sigma_l^{(i)}) = \text{MLP}(x^{(i)})$.The joint distribution $p_\theta(x^{(i)}, z^{(i)}, l^{(i)})$ is obtained for each modality via the Gaussian, log Normal prior and likelihoods in equation 2 and equation 3 for ST and SC, respectively.

Specifically, we would like to estimate parameters ψ and θ to maximize the ELBO for each modality, which is equivalent to minimize the loss function below

$$\mathcal{L}_{\text{VAE}}^{(i)} = \underbrace{-\mathbb{E}_{q_\psi(z^{(i)}|x^{(i)})}[\log p_\theta(x^{(i)}|z^{(i)})]}_{\text{Negative log likelihood}} + \underbrace{\text{KL}(q_\psi(z^{(i)}|x^{(i)})\|p(z^{(i)}) + \text{KL}(q_\psi(l^{(i)}|x^{(i)})\|p(l^{(i)})}_{\text{Regularization term}},$$

where the first term denotes modality-specific negative log-likelihood for ST and SC in (3) and (2), respectively. The regularization term is the Kullback–Leibler (KL) divergence, which encourages the distribution of the variational posteriors $q_\psi(z^{(i)}|x^{(i)})$ to be as close as possible to $p(z^{(i)})$ (a standard Gaussian) and the variational posteriors $q_\psi(l^{(i)}|x^{(i)})$ to be as close as possible to $p(l^{(i)})$.

A.3 Spatial Location Recovery Illustration Graph Shown in Figure 4

B EXPERIMENTS

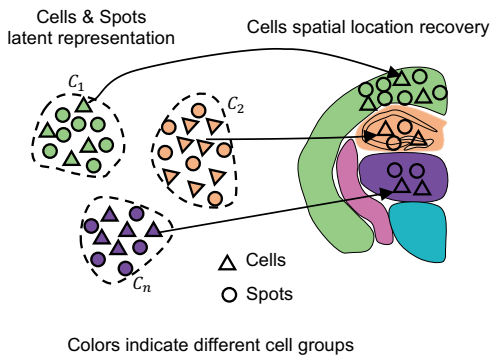

Figure 4: Spatial location recovery illustration graph. The left side is the latent representation for cells (triangles) and spots (circles), we use C_i and different colors to denotes different cell groups. Right: ST slices colored by cell-type. The arrow indicates the original ST location recovery process for SC.

B.1 Dataset Links and Summaries

Data availability Dataset links for spatial gene imputation: (1) seqFISH data on mouse organogenesis `https://crukci.shinyapps.io/SpatialMouseAtlas/` and the SC mouse embryo on E8.5 `https://doi.org/10.1038/s41586-019-0933-9`. (2) MERFISH data of spatial cell atlas of whole mouse brain `https://doi.org/10.1038/s41586-023-06808-9` and the SC data for all cortical and hippocampus region in the mouse brain `https://doi.org/10.1016/j.cell.2021.04.021`. (3) Mouse VISP: MERFISH, `https://github.com/spacetx-spacejam/data/`; Smart-seq, mouse primary visual cortex (VISp) in `https://portal.brain-map.org/atlases-and-data/rnaseq/mouse-v1-and-alm-smart-seq`. The data is summarized in Table 5.

Dataset	ST (spots × genes)	SC (cells × genes)	shared genes
Mouse embryo	19416 × 351	13298 × 29452	320
Mouse brain	6584 × 232	4633 × 31053	230
Mouse VISP	2376 × 268	14249 × 24408	242

Table 5: Dataset information.

Data pre-processing We follow the same data pre-processing steps for the Mouse embryo data as in the ENVI study. First, we filtered out mitochondrial genes and genes expressed in fewer than 1% of the cells. Cells with library sizes greater than 33,000 were removed. For the seqFISH mouse organogenesis data, we excluded spots with total expression over 600. Cell-type names were adjusted for consistency, as suggested in the ENVI paper. For the mouse brain single-cell data, we randomly selected around 500 cells from 10 sub-clusters in the cortical and hippocampal regions: 'L2 IT ENTl', 'L2/3 IT CTX', 'L4/5 IT CTX', 'L5 IT CTX', 'L6 CT CTX', 'L6b ENT', 'CA1-ProS', 'CA3-ve', 'CA3-do', and 'DG'. In the spatial transcriptomics (ST) mouse brain atlas data, we selected corresponding cell types from the 'Zhuang-ABCA-1.090' brain section and selected 232 highly variable genes in the regions of interest. For the Mouse VISP dataset, we applied the same filtering criteria to both SC and ST data, removing cells/spots with library sizes below 100 and genes with no expression.

B.2 Training and Parameter Settings for Baseline Methods

Naive baseline The SC and ST counts data are first normalized by library size, then followed by log transform and z-score normalization. Then we calculate the cosine similarity matrix between SC and ST. For a given spot the top 50 cells are retained to impute the unmeasured gene. Then we calculate the average normalized gene profile value in the top 50 cells and get the final imputed gene profile by multiplying the library size for that spot.

SpaGE We run SpaGE according to the official tutorial from `https://github.com/tabdelaal/SpaGE` and set the principle vector to 30 as suggested.

gimVI We follow the official tutorial from `https://docs.scvi-tools.org/en/1.0.0/tutorials/notebooks/gimvi_tutorial.html` to run gimVI with batch size 128, latent embedding dimension 10, and we train the model with 200 epochs.

Tangram For all three datasets, Tangram was run in cell mode as shown in `https://tangram-sc.readthedocs.io/en/latest/getting_started.html#running-tangram` and we set the learning rate to 0.1 with 2000 training epochs.

ENVI We run ENVI code as shown on github `https://github.com/dpeerlab/ENVI` with default parameter settings.

CeLEry we follow the offical CeLEry tutorial: `https://github.com/QihuangZhang/CeLEry/blob/main/tutorial/tutorial.md`. The raw gene expression was processed with log-transformed and z-score normalization. We train CeLEry on the seqFISH mouse embryo data with default parameter settings. Regarding the MERFISH VISP mouse brain dataset, we train the model with batch size 128, maximum epochs 100 and initial learning rate 10e-5.

scSpace We follow the tutorial on `https://github.com/ZJUFanLab/scSpace` and train scSpace with batch size 128, learning rate 0.001, and we set the number of epochs to 1000.

B.3 How to Derive the Calibration Plots

As Figure 10 shows, each gray line in the plot indicates gene-specific calibration plot, which is plotted as the actual observed frequency of the positive class (y-axis) against predicted probabilities (x-axis), to quantify the uncertainty for a specific imputed gene in predicting zero values. Afterward, every gene is fitted with ordinary least squares for each calibration curve, then the coefficient and intercept are calculated to quantitatively measure the goodness of the calibration in zero prediction for every gene in spatial transcriptomics. The average of the calibration curve (red dotted line) is plotted with the average intercept and slope across all genes, which is a summary for all the calibration curves for all genes in the dataset.

B.4 Statistical Test on Imputation, Calibration and Spatial Location Recovery

For spatial gene imputation task, our goal is to impute the expression of genes (G') that are absent in the ST dataset. To validate the imputation performance, we randomly held out a fraction of genes from ST and calculated evaluation metrics for each hold-out gene across all spatial spots. Specifically, we conducted 5-fold cross-validation on three datasets, ensuring that all imputed genes were systematically evaluated. The random split seed was recorded and applied consistently for all other methods for fair comparisons. The results reported in Tables 1, 2 represent the average performance across all imputed genes. Statistical significance was assessed using the Mann-Whitney U test, comparing our model's imputation results with those of other methods across all imputed genes.

For the results of the latent integration and spatial location recovery tasks in Table 3 and 4, each cell/spot in the latent embedding space was evaluated to determine whether its neighbors belonged to the same cell type, as visualized in the UMAP. Statistical tests were conducted by comparing kNN purity scores across all the cells/spots between our model and the baseline methods.

For the bASW score, which measures global integration between two modalities, the value is calculated as the average absolute silhouette width per cell type. As this metric yields a single value summarizing global integration, no statistical test was performed for this measure.

C RESULTS

C.1 Ablation Study (Table 6) Demonstrating the Efficacy of the DCA Module

	Mouse brain						
	Pearson↑	Spearman↑	Kendall tau↑	RMSE↓	AUC↑	Calib Slope(1)	Calib Intercept(0)
w/o DCA	0.298	0.250	0.199	1.712	0.713	**0.998**	0.036
our	**0.315**	**0.261**	**0.209**	**1.690**	**0.722**	1.033	**0.019**

Table 6: Spatial gene profile imputation and calibration results for DCA module ablation study. The text in **bold** demonstrates optimal performance.

C.2 Ablation Study (Figure 5 and Table 7) Demonstrating the Efficacy of the Cross-Modal and Contrastive-Pair Construction

	Mouse brain						
	Pearson↑	Spearman↑	Kendall tau↑	RMSE↓	AUC↑	Calib Slope(1)	Calib Intercept(0)
w/o sampling	**0.315**	0.256	0.204	1.758	0.716	0.907	0.093
our	**0.315**	**0.261**	**0.209**	**1.690**	**0.722**	**1.033**	**0.019**

Table 7: Spatial gene profile imputation and calibration results for cross modal and contrastive pair sampling ablation study. The text in **bold** demonstrates optimal performance.

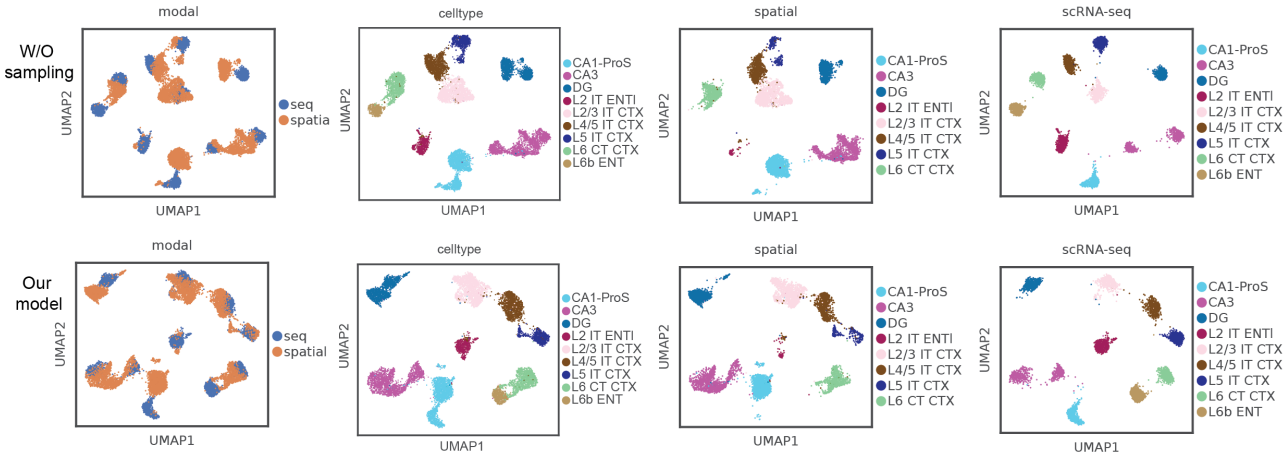

Figure 5: Ablation study for cross modal and contrastive pair sampling. Top panel: without cross-modal and contrastive pair construction; Lower panel: our model with sampling. From left to right: UMAP visualization for the two modalities: seq (blue) for SC and spatial (orange) for ST; UMPA visualization for the two modalities' embedding colored by cell types; UMPA visualization for ST in latent space colored by cell types; UMPA visualization for SC in latent space colored by cell types.

C.3 Ablation Study (Figure 6 and Table 8) of Training Spatial Location Recovery Jointly with the Full Reconstruction Loss

	Mouse embryo			
	knn5↑	knn10↑	knn20↑	knn50↑
with joint training	0.132	0.132	0.134	0.138
our	**0.290**	**0.291**	**0.293**	**0.291**

Table 8: Spatial location recovery performance for joint training position recovery with the reconstruction loss ablation study. The best results are in **bold**.

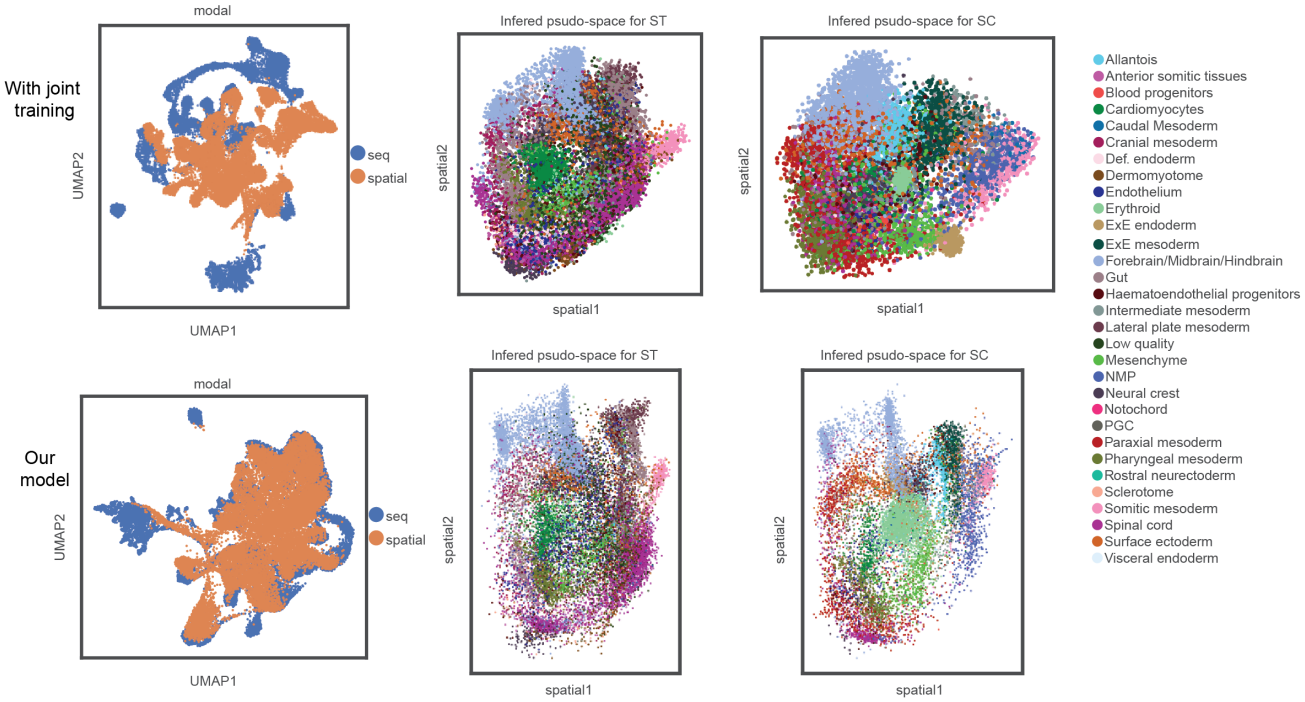

Figure 6: Ablation study for joint training position recovery with the reconstruction loss. Top panel: with joint training; Lower panel: our model without joint training. From left to right: UMAP visualization for the two modalities: seq (blue) for SC and spatial (orange) for ST; The predicted spatial location for ST; Inferred psudo-space for SC.

C.4 Imputation Performance Comparison (Table 9) between SpaGE and Our Model.

We conducted experiments with SpaGE on the Mouse Embryo and Mouse VISP datasets. Notably, for the Mouse brain dataset, we encountered an error related to "NA" values during PCA computation in SpaGE and were unable to generate valid results for this dataset. For consistency, we used the same gene splits as in our model and applied 5-fold cross-validation to evaluate spatial gene imputation performance. Additionally, we reported two versions of the correlation metrics for SpaGE: *i*) SpaGE log norm: Calculated using imputed gene expression values against log- and library-transformed counts, following the SpaGE official tutorial. *ii*) SpaGE: Calculated using imputed values against the raw counts, consistent with the evaluation approach for our model and other SOTA methods.

C.5 The Effect of Different Temperature Parameter τ (Table 10) in Contrastive Learning

In our experiments, we set $\tau = 1$ by default for all settings (training and prediction). In the following, we conducted additional experiments on the Mouse Brain dataset using different values of τ ($\tau = 0.5$, $\tau = 1$, and $\tau = 2$) to investigate its impact on the final model performance. The results are summarized in Table 10.

	Mouse embryo				Mouse VISP			
	Pearson	Spearman	Kendall tau	RMSE	Pearson	Spearman	Kendall tau	RMSE
SpaGE log norm	0.225	0.172	0.135	**0.457**	0.225	0.179	0.138	**1.313**
SpaGE	0.255	0.179	0.145	1.181	0.238	0.184	0.146	8.186
Ours	**0.405**	**0.336**	**0.267**	1.169	**0.386**	**0.345**	**0.273**	8.029

Table 9: Spatial gene imputation performance on Mouse embryo and Mouse VISP datasets for SpaGE and our model. The best results are highlighted in **bold**.

	Mouse brain						
	Pearson↑	Spearman↑	Kendall tau↑	RMSE↓	AUC↑	Calib Slope(1)	Calib Intercept(0)
τ=1	0.315	0.261	0.209	1.690	0.722	1.033	0.019
τ=0.5	0.318	0.262	0.209	1.690	0.722	0.989	0.049
τ=2	0.321	0.259	0.207	1.710	0.718	0.877	0.130

Table 10: Spatial gene profile imputation and calibration results for different τ in mouse brain dataset.

From these results, we observe that varying τ has a relatively minor impact on the model performance. While a higher value of τ ($\tau = 2$) slightly improves the Pearson correlation, it leads to small declines in the Spearman and Kendall Tau metrics, as well as the calibration performance. Similarly, a lower τ ($\tau = 0.5$) shows negligible changes overall. This suggests that the model is relatively robust to different τ settings within this range.

C.6 The Relationship between Imputation Performance, Input Discrepancies, and Latent Embedding Variance.

The relationship between input gene sparsity and imputation performance The sparsity level of gene expression varies significantly across individual genes in both scRNA-seq (SC) and spatial transcriptomics (ST). In our current study, we randomly split the common gene set into training and test sets. However, stratified sampling based on gene sparsity levels could serve as an alternative strategy to control the variance of the input features between the two modalities during training and testing.

To better understand the impact of sparsity and model performance, we analyzed imputation performance trends for the mouse cortex dataset (Figure 7). The results (shown in the figure below) suggest that genes with low sparsity and small discrepancy in sparsity levels between SC and ST tend to achieve better imputation performance.

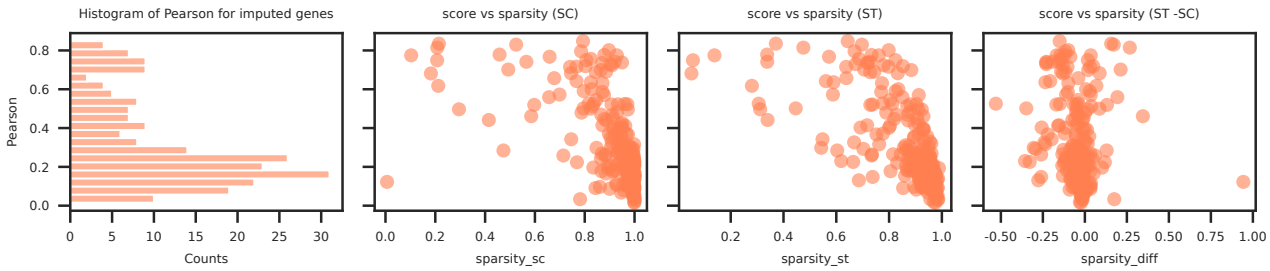

Figure 7: From left to right: histogram of Pearson correlation efficient for the imputed genes in Mouse brain data; the dot plot between Pearson correlation values and sparsity levels for the genes in SC; the dot plot between Pearson correlation values and sparsity levels for the genes in ST; the dot plot of Pearson correlation values and the difference sparsity levels between SC and ST.

Relationship between variance in the latent embeddings and predictive uncertainty In the following, we analyzed the relationship between the average latent embedding variance and predictive uncertainty in the mouse brain dataset, from two perspectives:

i) Imputation Performance *vs.* Latent Embedding Variance (Figure 8). For each spatial spot, we computed the average Gaussian variance vector, $\sigma_z^s \in \mathbb{R}^d$, under the assumption that the latent distribution is spherical,

this average variance value effectively represents the variational variance parameter of a given spot. Then we assessed its relationship with the imputation performance (*e.g.*, Pearson correlation in this case). As shown in the left panel of the figure below, the Pearson correlation decreases as the latent distribution variance increases, suggesting that a larger latent distribution variance indicates a less reliable imputation result for a given spot.

ii) Predicted Variance *vs.* Latent Embedding Variance (Figure 8). We analyzed the relationship between the average predicted variance (calculated using the negative binomial estimate for all genes in a spot) and the average latent embedding variance of the spot. Contrary to initial expectations, a negative relationship is observed ($R^2 = 0.11$, $p - value = 1.01 \times 10^{-171}$), where spots with a higher latent variance tend to have a lower predictive variance for their genes. This counterintuitive finding suggests that latent variance may not only capture representation uncertainty but could also act as a compensatory mechanism where higher latent variance reduces predictive uncertainty through learned regularities in the model.

These findings highlight the nuanced relationship between latent variance and model performance. The negative correlation between latent variance and imputation performance (left panel) suggests that reducing latent variance, for example, via regularization techniques or variance-aware objectives, could improve imputation accuracy. However, the negative correlation with predictive variance (right panel) opens an intriguing avenue for further exploration: latent variance may be stabilizing the predictive variance through its interaction with the learned latent space.

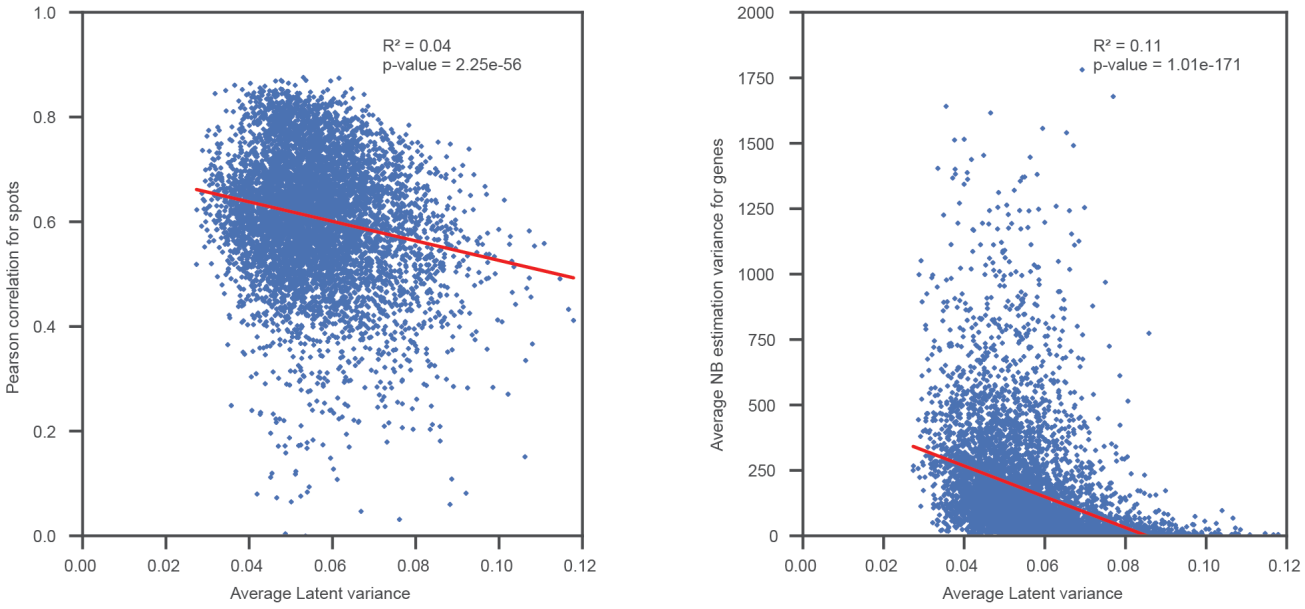

Figure 8: Left panel: the dot plot between imputation performance and latent embedding variance. Right panel: the dot plot between predicted gene variance and latent embedding variance.

C.7 Runtime and Hardware Usage (CPU/GPU) between Our Model and The Baseline Methods (Table 11)

The experiments were executed on a machine with Intel(R) Xeon(R) Gold 5320 CPU @ 2.20GHz, 128 RAM, and Nvidia Ada 5000 GPU.

	Mouse embryo			Mouse brain			Mouse VISP		
	Running time (min)	CPU (GB)	GPU (GB)	Running time (min)	CPU (GB)	GPU (GB)	Running time (min)	CPU (GB)	GPU (GB)
Naive	3.05	11.8	0	1.1	2.4	0	0.16	8.4	0
SpaGE	0.2	3.1	0	NA	NA	NA	0.1	4.7	0
gimVI	8.47	2.1	0.39	3.6	2.1	0.39	9.78	1.78	0.39
Tangram	3.7	5.4	7.04	0.52	1.6	1.35	0.62	4.2	1.35
ENVI	3.15	4.3	23.95	2.74	3.8	23.95	2.82	7.9	23.95
Ours	134	19.8	5.3	11	4.6	1.5	25	9.6	1.6

Table 11: Running time and hardware resource usage between our model and the baseline methods.

C.8 Boxplot for Spatial Gene Profile Imputation Benchmark (Figure 9) and Uncertainty Estimation (Figure 10, 11)

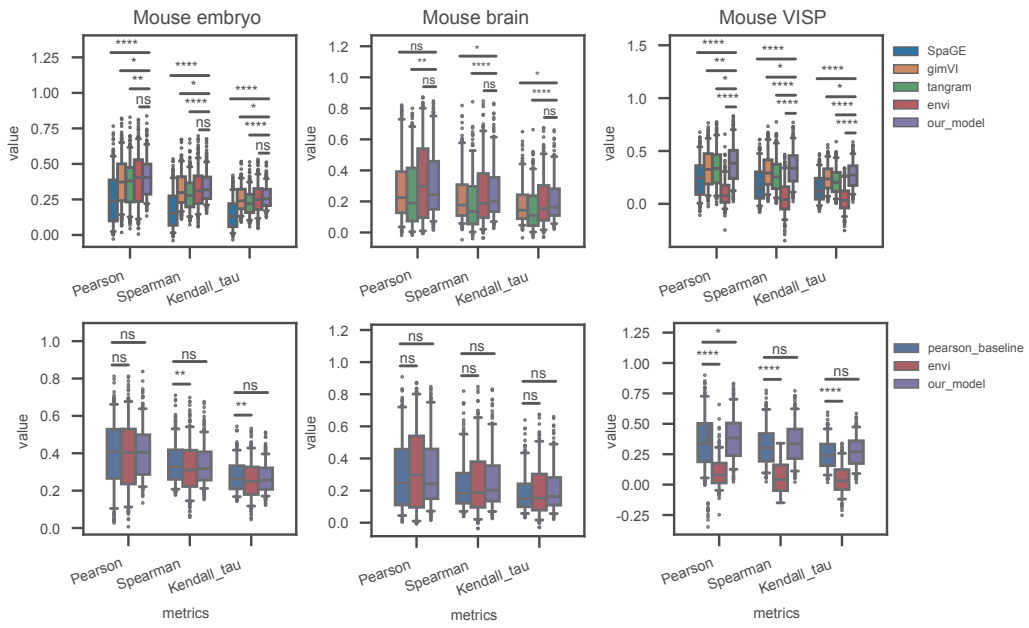

Figure 9: From left to right: Mouse embryo, Mouse brain and Mouse VISP dataset. Top row: imputation benchmark results for our methods *vs.* SpaGE, gimVI, Tangram and ENVI; Bottom row: imputation benchmark results for naive baseline *vs.* our method and ENVI. Statistical significance is indicated by stars (*), with "ns" denoting no significant difference.

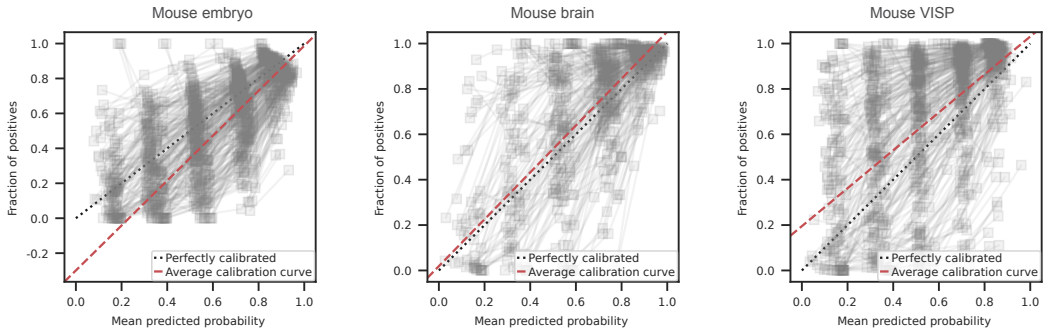

Figure 10: From left to right: calibration plot for genes in Mouse embryo, Mouse brain and Mouse VISP data. In each plot, every single gray line represents the calibration plot for an individual gene. The red dashed line is the averaging overall calibration curve for all the genes. The 45 degree black dash line indicates perfect calibration.

C.9 UMAP Visualization of Cross-Modal Latent Representation Results (Figure 12, 13).

C.10 Spatial Location Ground Truth Reference in ST in Mouse Embryo and Mouse Brain Datasets (Figure 14). Spatial Location Recovery Visualization for CeLEry, scSpace, and Our Method (Figure 15).

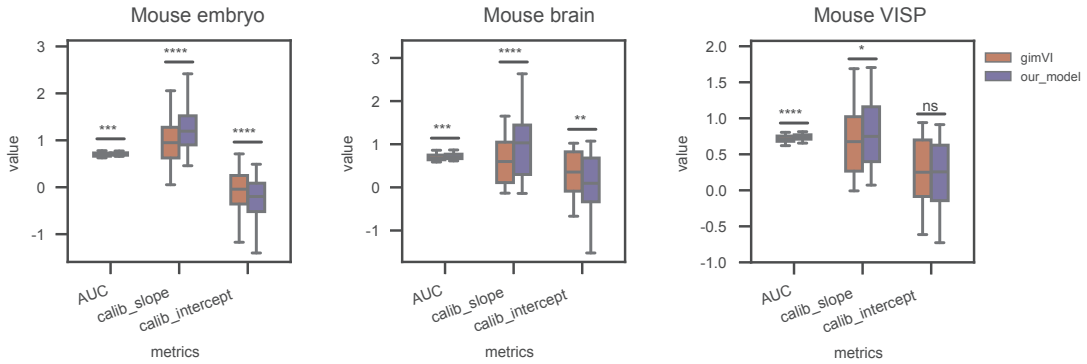

Figure 11: From left to right: calibration performance boxplot for our method and gimVI in Mouse embryo, Mouse brain and Mouse VISP data. Statistical significance is indicated by stars (*), with "ns" denoting no significant difference.

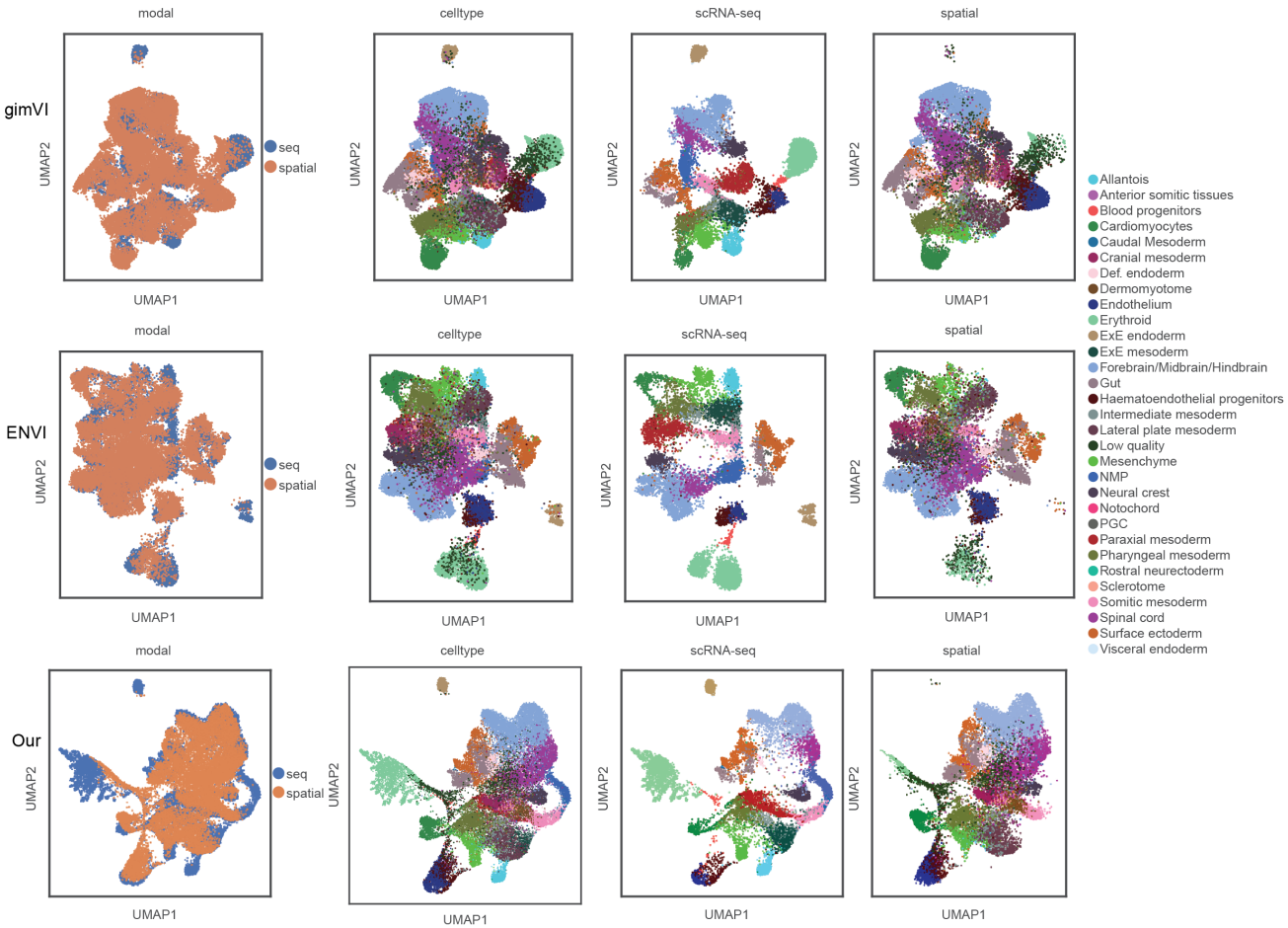

Figure 12: Cross-modal latent integration results for gimVI (top row), ENVI (middle row) and our method (bottom row) in Mouse embryo dataset. From left to right: the first two columns are the integrated latent representation colored by modality and cell types, respectively; the latter two columns are UMAP visualization for each modality individually, which is colored by cell types.

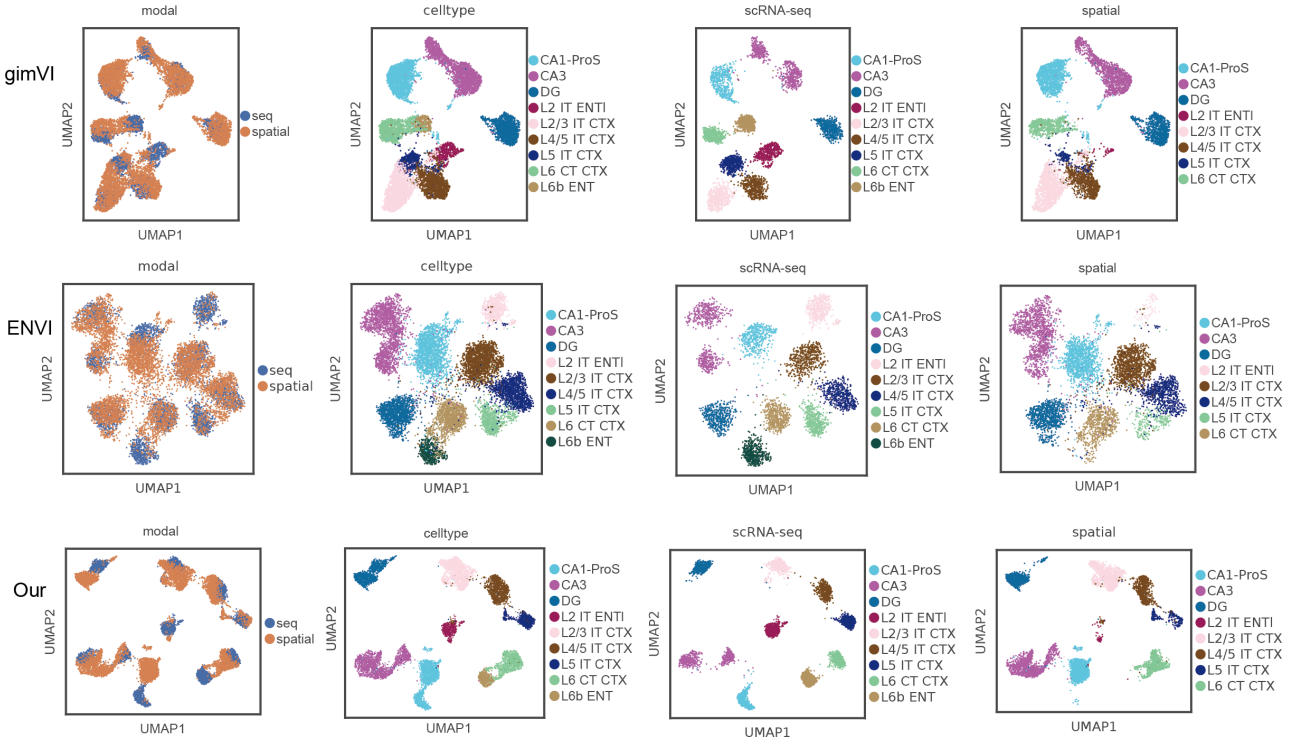

Figure 13: Cross-modal latent integration results for gimVI (top row), ENVI (middle row) and our method (bottom row) in Mouse brain dataset. From left to right: the first two columns are the integrated latent representation colored by modality and cell types, respectively; the latter two columns are UMAP visualization for each modality individually, which is colored by cell types.

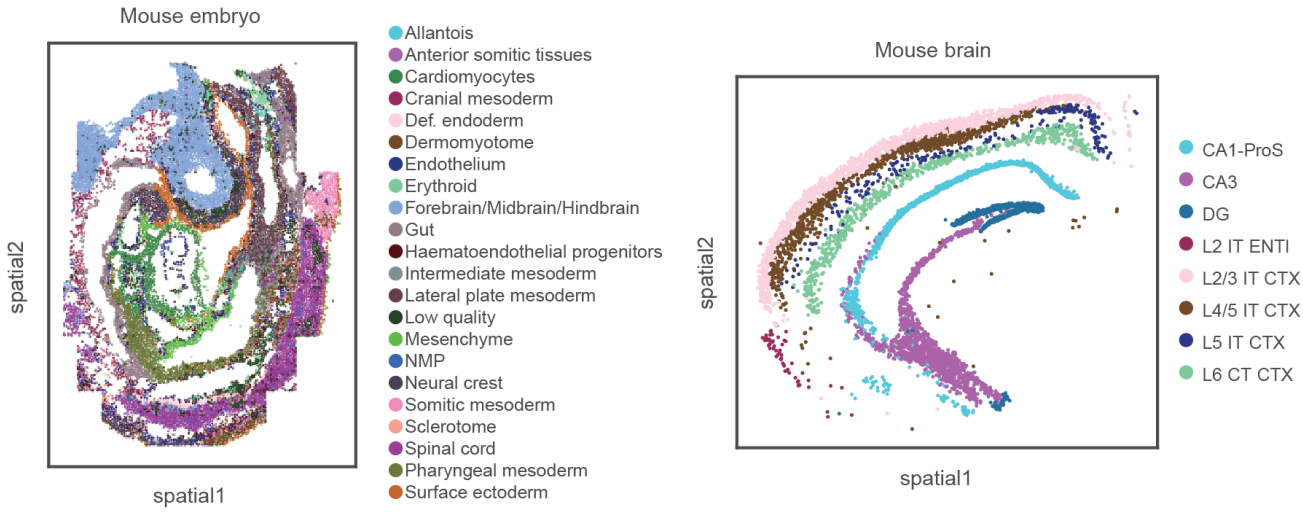

Figure 14: Spatial location ground truth reference in Mouse embryo and mouse brain datasets.

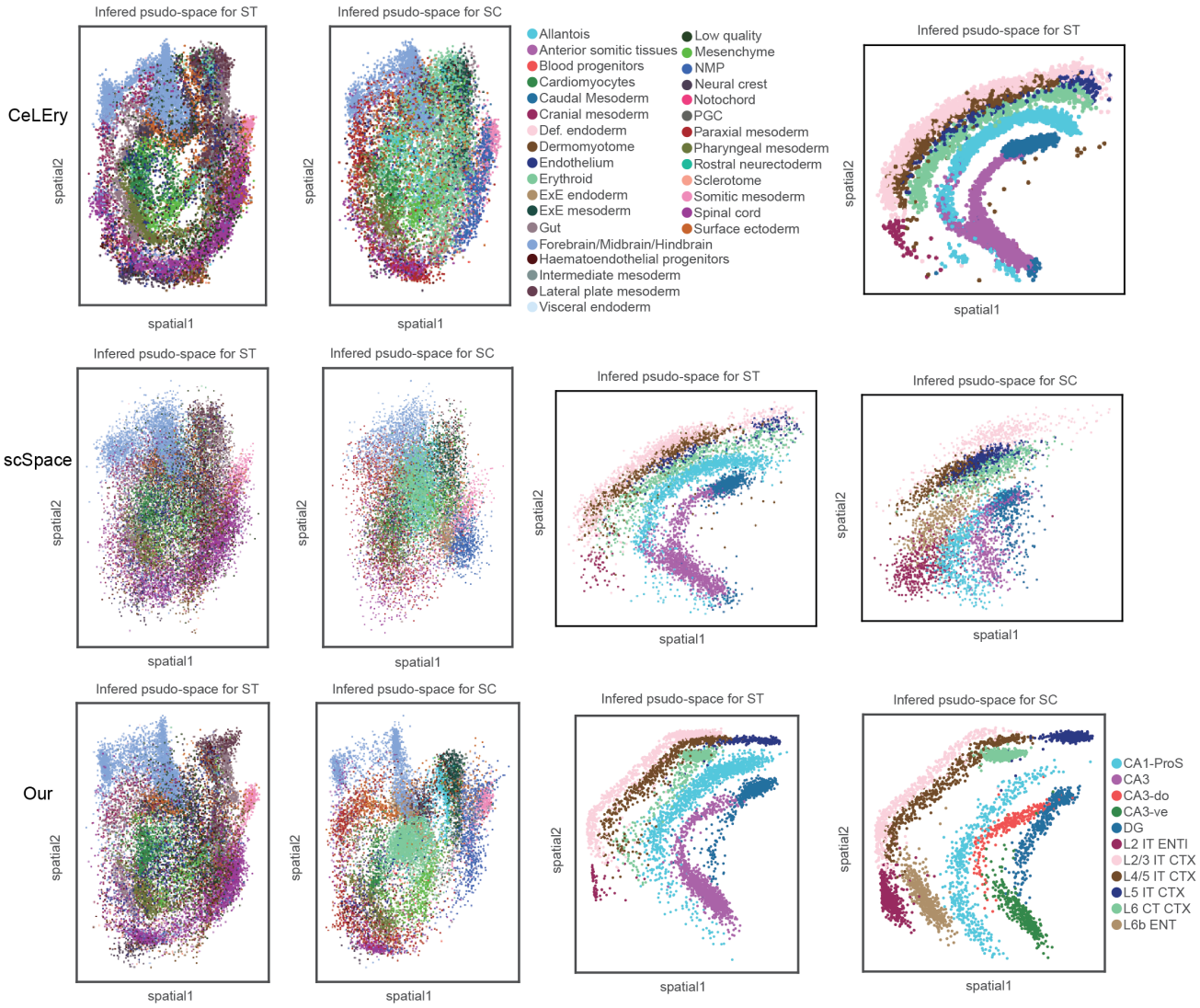

Figure 15: Spatial location recovery results for CeLEry, scSpace and our method in Mouse embryo and Mouse brain datasets. Left two columns: the predicted spatial location for ST and inferred pseudo-space for SC in Mouse embryo data; right two columns: the predicted spatial location for ST and inferred pseudo-space for SC in Mouse brain data.