Supplementary information for 'Deciphering spatial domains from spatial multi-omics with SpatialGlue'

Simulation data generation

To provide data samples with ground truth for evaluation, we followed the approach outlined by Townes et al. [1] to use nonnegative spatial factorization to simulate spatial multi-omics data. We employed the 'ggblocks' model from Townes et al. to generate expression matrices of different modalities. For modality 1, we generated a spatial gene expression matrix of 1,296 cells x 1,000 genes with the zero-inflated negative binomial (ZINB) distribution, featuring 4 distinct factors. Similarly, we generated a spatial protein expression matrix (modality 2) with negative binomial (NB) distribution, dimensions of 1,296 cells x 100 proteins, and featuring 4 distinct factors. The simulation approach recapitulates the ZINB and NB distributions of spatial transcriptomics and proteomics respectively and matches the cells from the two modalities. We also added Gaussian distributed noise to both modalities to better mimic real-world scenarios. To increase statistical analysis power, we generated 5 simulation datasets with different parameters. The summary statistics of the 5 simulation datasets are shown in Table S3.

Benchmarking methods

To evaluate the performance of SpatialGlue, we compare it with 10 state-of-the-art methods, including 7 single-cell multi-omics data integration methods, Seurat [2], totalVI [3], MultiVI [4], MOFA+ [5], MEFISTO [6], scMM [7], and StabMap [8], and 3 single-modal methods, SpaGCN [9], STAGATE [10], and GraphST [11].

All of benchmarking methods were executed based on their provided vignettes. For Seurat's data pre-processing of the RNA modality, 2,000 and 3,000 highly variable genes were selected for log normalization for the RNA & protein and RNA & ATAC (histone) data, respectively. The dimensions of feature reduction were set to 30 and 18 for the RNA and protein modalities, respectively. For the RNA & ATAC (histone) data, the dimensions of feature reduction were set to 10 for both the RNA and ATAC (histone) modalities. For MOFA+ and MEFISTO, the top 2,000 and 5,000 highly variable genes and peaks were chosen for normalization for the RNA and ATAC (histone) modality data, respectively. The number of factors was set to 10. For totalVI and MultiVI, we employed the scVI package (version 1.0.2) for data integration. The input dataset was preprocessed using the standard SCANPY workflow. Specifically, the top 4,000 highly variable genes were selected for log-normalization when implementing totalVI tool. For MultiVI, genes and peaks expressed in fewer than 1% pixels were eliminated for the RNA & epigenome (ATAC, histone) data. We need to highlight that totalVI was designed only for CITE-seq. scMM was executed with its default settings. The epochs, batch size, and learning rating were set to 50, 32, and 0.0001, respectively. Following the tutorial provided by the original paper, we ran StabMap using default settings. SpaGCN, STAGATE, and GraphST are deep learning models designed for single-modal spatial transcriptomics data. All three models were employed based on the tutorials provided. To adapt these methods to spatial multi-omics data, we concatenated the pre-processed expression matrices of the RNA and protein/ATAC/histone data as input to obtain latent representations. For totalVI, MultiVI, scMM, SpaGCN, STAGATE, and GraphST, after model training, we extracted the latent representations to perform clustering with the 'mclust' algorithm [12].

Downstream analyses

Spatial clustering. Taking the expression data of different omics modalities as input, SpatialGlue outputs an integrated representation of spots/cells. With the output representations as input, we applied the 'mclust' algorithm [12] to identify spatial domains. We tested different numbers of clusters to select the clustering that best capture the known biological structures and/or cell types.

DEG analysis. After obtaining the clustering labels, differential expressed gene (DEG) analysis was performed on the identified clusters using Seurat v4.0 [2] to identify differentially expressed genes, proteins, or peaks. Similarly, tSNE and UMAP plots were generated using the integrated representations for visualization.

Signac. We first performed log-normalization followed by data scaling on the 'SCT' assay using the Seurat package. To find differentially expressed genes, we used the 'FindAllMarkers' function with the following parameter settings: logfc.threshold = 0.1, min.pct = 0.1 and 'wilcoxon' test. We then ran Term frequency-inverse document frequency (TF-IDF) normalization, FeatureSelection and RunSVD (Singular value decomposition), followed by data scaling on the CUT&Tag assay using Signac v1.8.0 [13]. We estimated the Gene Activity scores and used the SpatialGlue's clustering results to identify differentially expressed genes using the Gene Activity Scores using the 'FindAllMarkers' function with the following parameters settings: logfc.threshold = 0.25, min.pct = 0.25 and 'wilcoxon' test.

ArchR. To estimate the differentially expressed peaks, we employ the ArchR package v1.0.2 [14]. We first created 'arrow' files using the parameters: minFrags = 0, maxFrags = 1e+07, tile_size = 5000, and 'the mm10' genome. We computed the dimensionality reduced space via IterativeLSI with dims = 1:30 and performed clustering using the standard Seurat neighborhood detection method via addClusters, followed by UMAP via the 'addUMAP' function. We then prepare the spatially resolved ATAC object as follows: the spatial information was integrated using Seurat's 'Read10X_image' function to create a 10x Genomics Visium object named image containing all the spatial folder information. We then filtered out the off-tissue pixels in both the image object and in the ArchR object. The gene score matrix containing all gene accessibility scores and metadata including the computed SpatialGlue clusters was then extracted, and the image object is added.

We next generated a reproducible peak set in ArchR using the 'addReproduciblePeakSet' function and called the peaks using MACS2 [15]. The differentially expressed peaks were then identified for the SpatialGlue's clusters in the 'PeakMatrix' with the 'getMarkerFeatures' function. Marker genes with differential gene scores were also computed from the 'GeneScoreMatrix' using the same function. Finally, we computed the linkage between genes and peaks using the 'addPeak2GeneLinks' function with the 'Iterative LSI' reductions and 'GeneScoreMatrix' values. We ran the 'addPeak2GeneLinks' function with the following settings: corCutOff = 0.45 and resolution = 1,000.

Evaluation metrics

To evaluate the data integration performance of the model, we used eight quantitative metrics, of which six are supervised metrics (AMI, NMI, ARI, homogeneity, mutual information, and V-measure), and two unsupervised metrics (Jaccard similarity and Moran's *I* score). The supervised metrics were computed using the scikit-learn [16] package in Python.

Jaccard similarity. Similar to the metric employed by Ghazanfar et al. [8], for spot i, we separately extract the sets N_{im} and N_{ie} of size k (default 50) containing the nearest spots in the m -th modality and embedding space, that is, $N_{im} =$

$\{set\ of\ neighbors\ of\ feature\ space\ m,\ s.t.\ rank(D(Z_{im}, Z_{jm})) \leq k\}$, where $D(a,b)$ is the Euclidean distance of vectors a and b. The Jaccard similarity is thus:

$J_i = Jaccard(N_{im}, N_{ie}) = \frac{|N_{im} \cap N_{ie}|}{|N_{im} \cup N_{ie}|}$.

A larger value of J_i means greater similarity between the integrated representation and the m-th modality data.

Moran's I score. Moran's I score was calculated using the Squidpy package (Palla et al.[17]). Briefly, given a feature (gene or label) and the spatial location of observations, Moran's I score assesses whether the pattern expressed is clustered, dispersed, or random (Getis et al.[18]). Specifically, Moran's I is defined as:

$$I = \frac{n}{W} \frac{\sum_i^n \sum_j^n w_{i,j} z_i z_j}{\sum_{i=1}^n z_i^2}$$

where z_i is the deviation of the feature from the mean ($x_i - \tilde{X}$), $w_{i,j}$ is the spatial weight between observations, n is the number of spatial units, and W is the sum of all $w_{i,j}$. A higher value of Moran's *I* score corresponds to a more centralized spatial pattern.

Neighborhood enrichment and co-occurrence. To assess the spatial relationships between clusters, we calculate neighborhood enrichment and co-occurrence scores with the Squidpy [17] package. First, we applied the 'mclust' algorithm on the output representation of SpatialGlue to obtain spatial clusters. With the spatial clusters as input, we calculated the co-occurrence score of each cluster using Squidpy. For neighborhood enrichment analysis to evaluate spatial autocorrelations of clusters, we used the spatial clusters and coordinates as input to Squidpy.

Ablation studies

Here we performed a series of ablation studies to illustrate the impact of different components in the SpatialGlue model on performance. We first considered the use of attention (A) over concatenation (C) in integrating information. We created three variants of SpatialGlue (AC, CA, CC). The variants CA and CC clearly showed deterioration in capturing the original data and AC was the closest in performance to SpatialGlue (Figure S5c). This was also reflected in the computed supervised metrics (Figure S5d). We next demonstrated the importance of spatial information by feeding in data without spatial information. Without spatial information, the output of the variant was much noisier than that of SpatialGlue (Figure S5e). The metrics also showed that this variant performed worse (Figure S5f). Finally, we fed the original data into SpatialGlue and found no performance difference (Figure S5g,h). This suggested that PCA pre-processing does not negatively impact performance while offering the benefit of reduced data dimension and hence reduce memory requirements and speed up subsequent computation.

Finally, we also tested SpatialGlue alongside single-modal methods with simple data concatenation of simulated and experimentally acquired data. With simulated data, GraphST and SpatialGlue achieved similar performance while STAGATE and SpaGCN's outputs were noisier (Figure S6b,c). The second test employed the P22 mouse brain data with RNA-Seq and ATAC-Seq modalities (Figure S6d,c,e). STAGATE's output showed high levels of smoothing (Moran's I score) but lowest similarity to the data modalities (lowest Jaccard Similarity). SpaGCN achieved the lowest Moran's I score and also failed to delineate the cortex layers. GraphST and SpatialGlue obtained similar Moran's I score but SpatialGlue was the overall best in terms of Jaccard Similarity. Visually, SpatialGlue was also able to capture the cortex layers more accurately.

Sensitivity to parameters

Using the simulated data, we tested SpatialGlue's sensitivity to parameter changes, namely the number of neighbors *k*, the number of PCs, and the number of GNN layers. The performance of SpatialGlue clearly varies significantly with *k* increasing both visually and in terms of metrics (Figure S7a,b). At relatively small values of *k* (3 to 6), we consider the performance loss to be tolerable. By default, we set *k* to 3. For number of PCs selected, the performance increases with the PCs count increasing but reduces from 25 to 50. The initial increase can be attributed to the additional PCs capturing more information while the poor performance at 50 is likely due to the higher dimension PCs containing noise instead. We also evaluated SpatialGlue's performance with the number of GNN layers ranging from 1 to 3. Our results showed SpatialGlue achieving the best performance with 1 GNN layer. Therefore, we use this value as the default in our model.

Reference

1. Townes, F. W. & Engelhardt, B. E. Nonnegative spatial factorization applied to spatial genomics. *Nat. Methods* **20**, 229–238 (2023).
2. Hao, Y. *et al.* Integrated analysis of multimodal single-cell data. *Cell* **184**, 3573-3587.e29 (2021).
3. Gayoso, A. *et al.* Joint probabilistic modeling of single-cell multi-omic data with totalVI. *Nat. Methods* **18**, 272–282 (2021).
4. Ashuach, T. *et al.* MultiVI: deep generative model for the integration of multimodal data. *Nat. Methods* **20**, 1222–1231 (2023).
5. Argelaguet, R. *et al.* MOFA+: A statistical framework for comprehensive integration of multi-modal single-cell data. *Genome Biol.* (2020) doi:10.1186/s13059-020-02015-1.
6. Velten, B. *et al.* Identifying temporal and spatial patterns of variation from multimodal data using MEFISTO. *Nat. Methods* **19**, 179–186 (2022).
7. Minoura, K., Abe, K., Nam, H., Nishikawa, H. & Shimamura, T. A mixture-of-experts deep generative model for integrated analysis of single-cell multiomics data. *Cell reports methods* **1**, 100071 (2021).
8. Ghazanfar, S., Guibentif, C. & Marioni, J. C. Stabilized mosaic single-cell data integration using unshared features. *Nat. Biotechnol.* (2023) doi:10.1038/s41587-023-01766-z.
9. Hu, J. *et al.* SpaGCN: Integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. *Nat. Methods* **18**, 1342–1351 (2021).
10. Dong, K. & Zhang, S. Deciphering spatial domains from spatially resolved transcriptomics with an adaptive graph attention auto-encoder. *Nat. Commun.* **13**, 1739 (2022).
11. Long, Y. *et al.* Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST. *Nat. Commun.* **14**, 1155 (2023).
12. Scrucca, L., Fop, M., Murphy, T. B. & Raftery, A. E. mclust 5: Clustering, Classification and Density Estimation Using Gaussian Finite Mixture Models. *R J.* **8**, 289–317 (2016).
13. Stuart, T., Srivastava, A., Madad, S., Lareau, C. A. & Satija, R. Single-cell chromatin state analysis with Signac. *Nat. Methods* **18**, 1333–1341 (2021).

14. Granja, J. M. *et al.* ArchR is a scalable software package for integrative single-cell chromatin accessibility analysis. *Nat. Genet.* **53**, 403–411 (2021).

15. Zhang, Y. *et al.* Model-based analysis of ChIP-Seq (MACS). *Genome Biol.* **9**, R137 (2008).

16. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

17. Palla, G. *et al.* Squidpy: a scalable framework for spatial omics analysis. *Nat. Methods* **19**, 171–178 (2022).

18. Getis, A. & Ord, J. K. The analysis of spatial association by use of distance statistics. Geogr. Anal. 24, 189–206 (2010).

Tables

Table S1 Experimental datasets used in the manuscript

Dataset	Name	Platform	Size (spots x genes/proteins/peaks)	Figure
Dataset1	Mouse spleen replicate1	SPOTS (RNA-protein)	2,568x32,285 2,568x21	Figures 3e-l, S24, S25a-c
Dataset2	Mouse spleen replicate2	SPOTS (RNA-protein)	2,768x32,285 2,768x21	Figure S25d-h
Dataset3	Mouse Thymus1	Stereo-CITE-seq (RNA-protein)	4,697x23,622 4,697x51	Figures 3a-d, S16, S20a, S21
Dataset4	Mouse Thymus2	Stereo-CITE-seq (RNA-protein)	4,253x23,529 4,253x19	Figures S17, S20b, S22
Dataset5	Mouse Thymus3	Stereo-CITE-seq (RNA-protein)	4,646x23,960 4,646x19	Figures S18, S20c, S23a
Dataset6	Mouse Thymus4	Stereo-CITE-seq (RNA-protein)	4,228x23,221 4,228x19	Figures S19, S20d, S23b
Dataset7	Mouse Brain RNA ATAC P22	Spatial-transcriptome-epigenome	9,215x22,914 9,215x121,068	Figures 2a-e, S12a,c,e
Dataset8	Mouse Brain RNA H3K4me3	Spatial-transcriptome-epigenome	9,548x22,731 9,548x35,270	Figure S14a-e
Dataset9	Mouse Brain RNA H3K27ac	Spatial-transcriptome-epigenome	9,370x23,415 9,370x104,162	Figures 2f-l, S12b,d,f, S13
Dataset10	Mouse Brain RNA H3K27me3	Spatial-transcriptome-epigenome	9,752x25,881 9,752x70,470	Figure S15
Dataset11	Human Lymph Node A1	10x Visium (RNA-protein)	3,484x18,085 3,484x31	Figures 1g-k, S8a-c, S9a, S10
Dataset12	Human Lymph Node D1	10x Visium (RNA-protein)	3,359x18,085 3,359x31	Figures S8d-k, S9b, S11,
Dataset13	Simulation 1	NSF (Townes et al., 2023)	1,296x1,000 1,296x100	Figures 1b-f, S1a
Dataset14	Simulation 2	NSF (Townes et al., 2023)	1,296x1,000 1,296x100	Figure S1b-f
Dataset15	Simulation 3	NSF (Townes et al., 2023)	1,296x1,000 1,296x100	Figure S2
Dataset16	Simulation 4	NSF (Townes et al., 2023)	1,296x1,000 1,296x100	Figure S3a-e
Dataset17	Simulation 5	NSF (Townes et al., 2023)	1,296x1,000 1,296x100	Figure S3f-j
Dataset18	Simulation 6 (triplet omics)	NSF (Townes et al., 2023)	1,296x1,000 1,296x100	Figure S4

Table S2 Application of methods on datasets

Datasets	Seurat	totalVI	MultiVI	MOFA+	MEFISTO	scMM	StabMap	SpatialGlue
Dataset1	√	√	√	√	√	√	√	√
Dataset2	√	√	√	√	√	√	√	√
Dataset3	√	√	√	√	√	√	√	√
Dataset4	√	√	√	√	√	√	√	√
Dataset5	√	√	√	√	√	√	√	√
Dataset6	√	√	√	√	√	√	√	√
Dataset7	√	$N.A.$	√	√	$N.A.$	√	√	√
Dataset8	√	$N.A.$	√	√	$N.A.$	√	√	√
Dataset9	√	$N.A.$	√	√	$N.A.$	√	√	√
Dataset10	√	$N.A.$	√	√	$N.A.$	$N.A.$	√	√
Dataset11	√	√	√	√	√	√	√	√
Dataset12	√	√	√	√	√	√	√	√
Dataset13	√	√	√	√	√	√	√	√
Dataset14	√	√	√	√	√	√	√	√
Dataset15	√	√	√	√	√	√	√	√
Dataset16	√	√	√	√	√	√	√	√
Dataset17	√	√	√	√	√	√	√	√

Table S3. Summary of simulation parameters. Here, “pi” denotes the zero-inflation probability of the ZINB (Zero-Inflated Negative Binomial) distribution, “nzprob_nsp” denotes the probability of a “one” (else zero) for nonspatial factors, and “bkg_mean” denotes the negative binomial mean for observations that are “zero” in the factors. “mean” and “std” are mean and standard deviation of the Gaussian distribution, respectively.

Dataset	Modality 1						Modality 2				
	ZINB			Gaussian		dimension	NB		Gaussian		dimension
	pi	nzprob_nsp	bkg_mean	mean	std		nzprob_nsp	bkg_mean	mean	std	
Simulation1	0.5	0.2	0.2	2	0.5	1,000	0.25	0.4	2	0.5	100
Simulation2	0.5	0.2	0.3	2	0.5	1,000	0.25	0.5	2	0.5	100
Simulation3	0.5	0.2	0.4	2	0.5	1,000	0.25	0.6	2	0.5	100
Simulation4	0.5	0.2	0.5	2	0.5	1,000	0.25	0.7	2	0.5	100
Simulation5	0.5	0.2	0.6	2	0.5	1,000	0.25	0.8	2	0.5	100

Table S4. Summary of technical specifications of different technologies.

Platform	Spatial resolution (μm)	Distance between spots (μm)	Image area size
10x Visium & SPOTS	55	100	6.5 x 6.5 mm
Stereo-seq	0.22	0.5	200 mm^2
Spatial-epigenome-transcriptome	20	-	50x50 or 100x100 grid