biology

MDPI

Article

DVCAlign: A Dual-View Graph Alignment Framework for Spatial Transcriptomics Data

Chunyang Meng †, Wei Cheng †, Xiang Ao, Yanbing Xiao and Yi Jiang *

School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; mengcy@cqupt.edu.cn (C.M.)
* Correspondence: jiangyi@cqupt.edu.cn
† These authors contributed equally to this work.

Simple Summary

Spatial transcriptomics lets researchers measure gene activity while keeping track of where each measurement was made in a tissue. Comparing several tissue sections can reveal which structures are shared and which change, but this comparison is difficult when sections overlap only partly, have irregular boundaries, or come from different developmental stages. We developed DVCAlign, a computational method that combines two kinds of information: the physical arrangement of measurement spots and similarities in their local gene activity. The method learns when these views agree and modulates the influence of candidate matches during cross-section alignment. We tested DVCAlign on brain, breast cancer and developing heart data. In the original six-execution human brain-section comparison, it gave the highest average agreement with known cortical layers among the methods evaluated. In partially overlapping mouse brain sections, it improved clustering agreement, although unmatched regions could still enter alignment supervision. Across breast cancer and heart sections, it preserved organized compartments and stage-related differences. These results suggest that DVCAlign can help researchers compare spatial transcriptomics sections while retaining the need to assess false correspondences and biological confounding.

Abstract

Integrating spatial transcriptomics sections is important for comparative analysis across samples, conditions and developmental stages, but remains difficult when sections differ in overlap, boundary geometry or developmental state. Existing approaches may rely primarily on transcriptional similarity, physical proximity or a shared latent representation, which can produce false matches in unmatched regions or blur local tissue boundaries. Here we present DVCAlign, a dual-view graph framework that separates spatial continuity from spatially local molecular similarity, fuses the two views at the spot level and applies confidence-weighted cross-slice supervision. Across four benchmark settings, DVCAlign yielded spatially structured embeddings across the evaluated partial-overlap, compartment-specific and stage-associated settings. In repeated dorsolateral prefrontal cortex evaluations, DVCAlign achieved a mean pooled adjusted Rand index of 0.633 across six executions, an absolute increase of 0.022 over STAligner, corresponding to a 3.6% relative improvement. In the mouse brain benchmark, DVCAlign achieved an adjusted Rand index of 0.441, the highest among the evaluated methods. In heterogeneous breast cancer sections, DVCAlign recovered a broadly organized compartment pattern across H1–H3, with marker-score contrasts supporting distinct transcriptional programs in H1. In a developmental human embryonic heart series, DVCAlign recovered stage-associated spatial organization while retaining structural and functional differences. These findings support embedding integra-

Check for updates

Academic Editor: Tianwei Yu

Received: 28 October 2026
Revised: 19 November 2026
Accepted: 25 November 2026
Published: 28 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

tion under the evaluated conditions; clustering agreement and visual correspondence do not independently establish anatomical registration accuracy.

Keywords: spatial transcriptomics; cross-slice integration; graph neural networks; spatial domains; multi-slice analysis

1. Introduction

Spatial transcriptomics (ST) enables gene-expression profiling to be interpreted in its native tissue context. With the development of array-based, sequencing-based and imaging-based ST platforms, analyses increasingly extend beyond individual sections to comparisons across samples, conditions and developmental stages [1–11]. These comparisons require methods that can identify spatial domains within each section and determine which structures are shared, altered or newly formed across related sections.

Section integration has therefore become an important computational task in comparative ST analysis. Unlike conventional single-cell RNA-sequencing integration [12], ST integration must account for both expression variation and tissue geometry. An effective method should reduce unwanted technical differences without erasing anatomical boundaries or genuine biological changes. This balance becomes difficult when sections have partial overlap, irregular compartment boundaries or developmental differences.

Existing approaches can be grouped according to the information used to define relationships between spots. General-purpose single-cell integration methods, including Harmony [13], Seurat [14], Scanorama [15] and BBKNN [16], operate primarily on transcriptional features and may merge anatomically distinct regions with similar expression profiles. Spatially informed methods such as STAGATE [17], SpaGCN [18], GraphST [19], SEDR [20] and PASTE and PASTE2 [21,22] model tissue geometry more explicitly, but their objectives often focus on within-section structure recovery or alignment under relatively strong correspondence. More recent multi-slice frameworks, including STAligner [23], SpaBatch [24], MaskGraphene [25], stClinic [26] and PRECAST [27], extend graph learning with triplet-based, contrastive or probabilistic supervision for cross-slice integration.

Despite these advances, many existing methods rely primarily on one dominant representation of spot relationships during alignment. Physical proximity can be unreliable near irregular boundaries or mixed compartments, whereas transcriptional similarity can connect anatomically distinct regions. A correspondence rule that works for adjacent sections may therefore become unreliable when overlap is partial, tissue compartments are irregular or biological state changes across slices. The unresolved problem is not simply how to increase cross-slice similarity, but how to strengthen correspondence where it is supported while preserving separation elsewhere.

Here we present DVCAlign (Dual-View Confidence-weighted Alignment), a dual-view graph alignment framework for structure-preserving ST integration. DVCAlign constructs complementary spatial and spatially local expression-similarity graphs, learns view-specific embeddings, combines them through a spot-wise fusion gate and applies confidence-weighted cross-slice supervision. We ask whether this design can improve correspondence when tissue organization is shared while retaining partial-overlap, compartment-specific and developmental differences. We evaluate DVCAlign across adjacent human dorsolateral prefrontal cortex (DLPFC) sections, partially overlapping mouse brain sections, heterogeneous breast cancer sections and a developmental series of human embryonic heart slices. Its contribution is the coordinated use of spatially constrained molecular neighborhoods, spot-wise fusion and cross-view-informed alignment weights; the individual graph, gate and triplet operators are established components.

Dedicated spatial registration and matching approaches, including STalign, SLAT, CAST, moscot and GPSA [28–32], address coordinate transformations, cross-sample matching or inference in a shared coordinate system. These goals differ from embedding integration for spatial-domain analysis. DVCAlign does not itself estimate a coordinate-deformation field or certify a unique anatomical counterpart for each spot. We therefore distinguish clustering agreement and biological label preservation from physical registration accuracy, and do not claim superiority over methods not directly benchmarked.

2. Materials and Methods

2.1. Overview of DVCAlign

DVCAlign is a dual-view graph framework for ST section integration and spatial-domain identification (Figure 1). Given slice-wise spatial coordinates and expression features, DVCAlign learns a fused low-dimensional representation for each spot while preserving within-slice tissue structure. The framework consists of five main modules: slice-wise preprocessing maps sections into a shared input space; dual-view graph construction defines complementary spatial and local-expression neighborhoods; a shared graph attention auto-encoder [33] learns view-specific embeddings; adaptive spot-wise fusion integrates the two views; and two-stage optimization introduces cross-slice alignment only after within-slice structure has been stabilized.

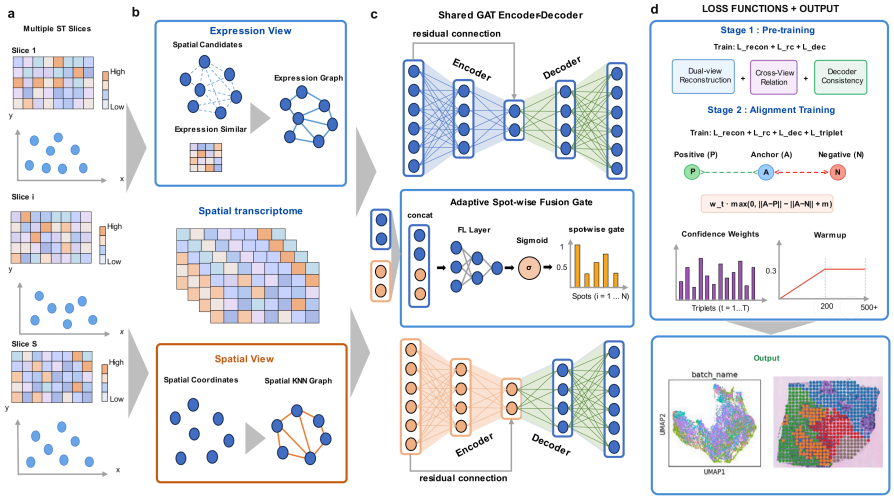

Figure 1. Overview of the DVCAlign framework. (**a**) Multi-slice ST sections provide spatial coordinates and gene-expression profiles. (**b**) A spatial view and a spatially local expression-similarity view are constructed for each section. (**c**) Shared graph-attention encoding and decoding with residual connections and spot-wise fusion produce the latent representation. (**d**) Two-stage optimization uses confidence-weighted triplet alignment to obtain integrated embeddings and downstream spatial domains.

2.2. Data Preprocessing

ST sections can differ in feature sparsity, gene coverage and spatial layout, which can make direct joint training unstable. DVCAlign therefore preprocesses each slice independently before mapping the slices into a shared feature space. Highly variable genes (HVGs) were selected on a slice-wise basis, using the seurat_v3 procedure on count data where available; a variance-based fallback was used for the heart workflows when seurat_v3 selection was not available. Expression counts were normalized to a library size of 10,000 per spot and log-transformed in Scanpy 1.9.1 [34]. Dataset-specific feature sets were then combined in a shared matrix, with missing genes represented consistently across slices, before graph construction and representation learning. The benchmark composition and

spot counts are summarized in Table 1, and dataset sources and accession information are listed in Table 2.

Table 1. Summary of spatial transcriptomics datasets used in DVCAlign.

Tissue	Platform	Slice ID	# of Spots †	Related Figures
Human dorsolateral prefrontal cortex (DLPFC)	10x Genomics Visium	151673 151674 151675 151676	3639 3673 3592 3460	Figure 2
Mouse brain	10x Genomics Visium	Sagittal-anterior (MA) Sagittal-posterior (MP)	2696 3353	Figure 3
HER2-positive human breast cancer	Spatial Transcriptomics (ST)	H1 H2 H3	614 604 511	Figure 4
Developing human embryonic heart	Spatial Transcriptomics (ST)	4.5-5 PCW: ST_Sample_4.5-5PCW_1 ST_Sample_4.5-5PCW_2 ST_Sample_4.5-5PCW_3 ST_Sample_4.5-5PCW_4	1137 1138 1136 1138 (registered array positions)	Figure 5
		6.5 PCW: ST_Sample_6.5PCW_1 ST_Sample_6.5PCW_2 ST_Sample_6.5PCW_3 ST_Sample_6.5PCW_4 ST_Sample_6.5PCW_5 ST_Sample_6.5PCW_6 ST_Sample_6.5PCW_7 ST_Sample_6.5PCW_8 ST_Sample_6.5PCW_9	1130 1152 1147 1133 1153 1153 1153 1148 1143 (registered array positions)	

† Heart coordinate files contain 14,861 registered array positions (4549 early-stage and 10,312 later-stage). The processed expression data used for Figure 5 and GO analysis contain 1753 analyzed spots (238 early-stage and 1515 later-stage); per-section analyzed counts are not inferred from coordinate counts.

Table 2. Dataset sources and accession information.

Related Figures	Dataset/Sections	Source or Accession	URL
Figure 2	Human DLPFC sections 151673-151676	spatialLIBD	https://research.libd.org/spatialLIBD/ (accessed on 19 November 2026)
Figure 3	Mouse brain serial sections, sagittal anterior and posterior	10x Genomics Mouse Brain Serial Section 2	https://www.10xgenomics.com/datasets/mouse-brain-serial-section-2-sagittal-anterior-1-standard (accessed on 19 November 2026); https://www.10xgenomics.com/datasets/mouse-brain-serial-section-2-sagittal-posterior-1-standard (accessed on 19 November 2026)
Figure 4	HER2-positive breast cancer sections H1–H3	HER2ST; Zenodo record 4751624	https://github.com/almaan/her2st (accessed on 19 November 2026); https://zenodo.org/records/4751624 (accessed on 19 November 2026)
Figure 5	Human embryonic heart sections, 4.5–5 and 6.5 PCW	Mendeley Data DOI: 10.17632/dgnysc3zn5.1; EGA: EGAD00001005468	https://data.mendeley.com/datasets/dgnysc3zn5/1 (accessed on 19 November 2026); https://ega-archive.org/datasets/EGAD00001005468 (accessed on 19 November 2026)

2.3. Dual-View Graph Construction

After preprocessing, DVCAlign represents each slice with two complementary within-slice graphs that define different neighborhoods for message passing. The first is a spatial graph that preserves local tissue geometry. The second is an auxiliary expression graph that refines neighborhoods by spatially local expression similarity. Keeping the two graphs separate allows spatial continuity and local molecular similarity to be encoded before cross-slice alignment is introduced.

Spatial graph. DVCAlign computes Euclidean distances from the spot coordinates stored in adata.obsm['spatial'] and connects neighboring spots with either a radius-based rule or a k-nearest-neighbor rule through Cal_Spatial_Net. The resulting edge list defines the neighborhood for graph propagation, and a corresponding adjacency matrix is stored for downstream processing. Because neighborhood scale depends on tissue layout and platform resolution, graph construction is selected according to the dataset rather than fixed globally.

Auxiliary expression graph. DVCAlign further constructs a second within-slice graph from spatially local expression similarity. For each slice, the candidate-neighbor pool is chosen from the total number of spots, an estimated cluster count obtained through unsupervised clustering and a fixed scaling factor of approximately 20, giving the setting used for aux_candidate_k. An expression-similar subset of these spatial candidates is retained as aux_expr_k neighbors. The spatial graph uses a radius-based rule selected so that the mean number of spatial neighbors per spot is close to aux_expr_k. Expression profiles are projected into principal component analysis (PCA) space and cosine similarity is computed within the local candidate pool. This preserves spatial locality while refining neighborhoods near boundaries or mixed regions. Reference annotations do not enter this neighborhood-selection heuristic; the estimated count is distinct from the reference category count. The chosen integer settings and reported dataset-specific configurations are documented in Supplementary Text S1.

2.4. Dual-View Graph Encoder and Decoder

Based on the two within-slice graphs, DVCAlign applies the same graph attention auto-encoder to the spatial view and the auxiliary view. The shared architecture keeps the two branches comparable while allowing them to learn view-specific embeddings from different neighborhood definitions. Each branch contains a graph attention encoder, a residual projection, and a weight-tied graph attention decoder.

Graph attention encoder. For a view-specific graph and normalized expression matrix $\mathbf{X}$, the encoder updates the representation of spot i by aggregating transformed neighbor features through graph attention. Here $\mathbf{X}$ denotes the input spot-by-feature matrix, $\mathbf{W}$ denotes a learnable layer transformation, and attention coefficients weight graph-neighbor contributions. Variable roles, operators and dimensions are summarized in Table S1. The update is

$$\mathbf{z}_i^{(k)} = \sigma\left(\sum_{j \in \mathcal{N}_i} \alpha_{ij}^{(k)} \mathbf{W}^{(k)} \mathbf{z}_j^{(k-1)}\right) \tag{1}$$

This update is computed separately on the spatial graph and the auxiliary expression graph, so the two branches preserve different within-slice neighborhood structure.

For edge (i,j), the attention coefficient is normalized over the neighbors of spot i:

$$\alpha_{ij}^{(k)} = \frac{\exp(e_{ij}^{(k)})}{\sum_{l \in \mathcal{N}_i} \exp(e_{il}^{(k)})} \tag{2}$$

In the implementation, source and target attention scores are computed separately, added at the edge level, passed through a sigmoid nonlinearity, and normalized by neighborhood softmax. The attention-score sigmoid and neighborhood softmax have distinct roles; the fusion-gate nonlinearity is also sigmoid. The hidden-layer activation in Equation (1) is the exponential linear unit (ELU) in the implementation.

$$e_{ij}^{(k)} = \mathrm{Sigmoid}\left(\mathbf{a}^{(k)\top}\mathbf{W}^{(k)}\mathbf{z}_i^{(k-1)} + \mathbf{a}^{(k)\top}\mathbf{W}^{(k)}\mathbf{z}_j^{(k-1)}\right). \tag{3}$$

A second graph attention layer then produces the latent embedding used for downstream fusion and alignment.

$$\mathbf{z}_i^{(L)} = \sum_{j\in\mathcal{N}_i} \alpha_{ij}^{(L)}\mathbf{W}^{(L)}\mathbf{z}_j^{(L-1)}. \tag{4}$$

Residual connection. Repeated graph aggregation may oversmooth local boundaries or sharp expression transitions. DVCAlign therefore adds a residual projection from the input feature space to the latent space:

$$\mathbf{h}_i = \lambda\mathbf{z}_i^{(L)} + (1-\lambda)\mathbf{W}_r\mathbf{x}_i \tag{5}$$

The final spatial-view embedding and auxiliary-view embedding are obtained by combining the graph-encoded term with the residual term using fixed weights of 0.7 and 0.3, respectively. The ratio was chosen empirically to emphasize graph-encoded features. Targeted residual-weight and gate-initialization sensitivity analyses used five training seeds per setting and are reported in Supplementary Text S1, Section S3 and Table S2; the retained defaults are not interpreted as universally optimal.

Graph attention decoder. To keep each view-specific embedding anchored to the measured transcriptional signal, DVCAlign reconstructs the input features with a weight-tied graph attention decoder. The decoder reuses encoder-derived attention information in the reverse pass, which reduces parameter redundancy and stabilizes reconstruction.

$$\hat{\mathbf{z}}_i^{(k-1)} = \sigma_{k-1}\left(\sum_{j\in\mathcal{N}_i} \hat{\alpha}_{ij}^{(k-1)}\hat{\mathbf{W}}^{(k)}\hat{\mathbf{z}}_j^{(k)}\right), \quad \sigma_1 = \mathrm{ELU}, \quad \sigma_0 = \mathrm{id} \tag{6}$$

The decoder output is treated as the reconstructed expression profile of each spot.

Dual-view forward propagation. Let the spatial graph and the auxiliary expression graph be denoted by G_s and G_a, with adjacency matrices $\mathbf{A}^{(s)}$ and $\mathbf{A}^{(e)}$, respectively. Given $\mathbf{X}$ and these two within-slice graphs, DVCAlign processes the two views in parallel and outputs two view-specific embeddings together with their reconstructions:

$$\mathbf{z}_i^{(s)}, \hat{\mathbf{x}}_i^{(s)} = \mathrm{GAE}(\mathbf{X}, \mathbf{A}^{(s)}) \tag{7}$$

$$\mathbf{z}_i^{(e)}, \hat{\mathbf{x}}_i^{(e)} = \mathrm{GAE}(\mathbf{X}, \mathbf{A}^{(e)}) \tag{8}$$

2.5. Adaptive Spot-Wise Fusion

The two encoder outputs contain complementary within-slice information, and their relative contributions may vary across spots. DVCAlign therefore uses a spot-wise fusion gate to adaptively combine the spatial-view embedding and the auxiliary-view embedding.

$$\alpha_i = \mathrm{Sigmoid}\left(\mathbf{w}_g^T[\mathbf{z}_i^{(s)} \| \mathbf{z}_i^{(e)}] + b_g\right) \tag{9}$$

For each spot, the concatenated two-view embedding is passed through a gate function to produce a mixing coefficient. The gate weights are initialized to zero and its bias to $\log(0.7/0.3)$, giving an initial spatial/expression contribution of 0.7/0.3 so that the fused representation remains spatially anchored at the start of training:

The fused embedding is then computed as a convex combination of the two view-specific embeddings. Unlike a fixed global weight, the learned gate permits spatial and auxiliary expression contributions to vary according to local embedding features. Component ablations support a contribution of adaptive fusion to integration performance under the evaluated settings (Figure S1).

$$\mathbf{z}_i = \alpha_i \mathbf{z}_i^{(s)} + (1 - \alpha_i)\mathbf{z}_i^{(e)} \tag{10}$$

2.6. Joint Optimization and Loss Functions

DVCAlign optimizes the fused representation using within-slice reconstruction and cross-view consistency before introducing cross-slice alignment. The training objective contains reconstruction loss, relation consistency loss, decoder consistency loss, and confidence-weighted triplet supervision.

Reconstruction loss. Each view-specific decoder reconstructs the input expression matrix from its latent embedding. The reconstruction loss is defined as the mean squared error between the input and reconstructed outputs in the two views.

$$\mathcal{L}_{\text{rec}} = \frac{1}{2Nd}\sum_{i=1}^{N}\left(\left\|\mathbf{x}_i - \hat{\mathbf{x}}_i^{(s)}\right\|^2 + \left\|\mathbf{x}_i - \hat{\mathbf{x}}_i^{(e)}\right\|^2\right), \quad d = \dim(\mathbf{x}_i) \tag{11}$$

Relation consistency loss. To preserve comparable cross-view structure without forcing the two embeddings to collapse, DVCAlign computes a cross-view similarity matrix after L2 normalization and penalizes its asymmetry with respect to the transpose. This term encourages reciprocal relational structure while allowing the two views to retain non-identical local information. In Equations (13) and (14), $\mathbf{Z}^s$ and $\mathbf{Z}^e$ denote the row-wise L2-normalized embeddings from Equation (12).

$$\widetilde{\mathbf{Z}}^s = \text{Norm}(\mathbf{Z}^s), \quad \widetilde{\mathbf{Z}}^e = \text{Norm}(\mathbf{Z}^e), \tag{12}$$

$$\mathbf{C} = \widetilde{\mathbf{Z}}^s (\widetilde{\mathbf{Z}}^e)^\top, \tag{13}$$

$$\mathcal{L}_{\text{rc}} = \frac{1}{N^2}\left\|\mathbf{C} - \mathbf{C}^\top\right\|_F^2. \tag{14}$$

Decoder consistency loss. Because the two decoders reconstruct the same underlying expression signal, DVCAlign further applies a smooth L1 loss between the two reconstructed outputs. This term limits branch-specific drift and stabilizes training on noisy or sparse slices.

$$\mathcal{L}_{\text{dec}} = \frac{1}{N}\sum_{i=1}^{N}\text{SmoothL1}(\hat{\mathbf{x}}_i^{(s)}, \hat{\mathbf{x}}_i^{(e)}) \tag{15}$$

Confidence-weighted triplet loss. Cross-slice alignment is introduced through triplet supervision on the fused embedding. Anchor-positive pairs are defined as cross-slice mutual nearest-neighbor (MNN) pairs [35]. When an anchor has multiple MNN candidates, the closest candidate in the current fused embedding is selected as the positive, and the negative is sampled from the same slice as the anchor. The triplet set is therefore updated from the current embedding rather than fixed before training.

To avoid assigning the same strength to all MNN pairs, DVCAlign estimates a confidence weight from cross-view support. For each anchor-positive pair, cosine similarity is computed separately in the spatial-view embedding and the auxiliary-view embedding. Pairs supported by both views receive larger weights than pairs supported mainly by only one view. These weights are relative consistency multipliers, not calibrated probabilities of correct matching, and do not impose hard rejection of unmatched anchors. Controlled

reliability and matched weighted-versus-uniform feedback diagnostics are reported in Figure S2.

$$\mathrm{sim}_{ap}^{(s)} = \frac{\mathbf{z}_a^{(s)} \cdot \mathbf{z}_p^{(s)}}{\|\mathbf{z}_a^{(s)}\|\|\mathbf{z}_p^{(s)}\|} \tag{16}$$

$$\mathrm{sim}_{ap}^{(e)} = \frac{\mathbf{z}_a^{(e)} \cdot \mathbf{z}_p^{(e)}}{\|\mathbf{z}_a^{(e)}\|\|\mathbf{z}_p^{(e)}\|} \tag{17}$$

Specifically, the confidence weight combines the average cross-view similarity of the pair with the agreement between the two similarities:

$$\mathrm{avg_sim}_{ap} = \frac{\mathrm{sim}_{ap}^{(s)} + \mathrm{sim}_{ap}^{(e)} + 2}{4} \tag{18}$$

$$\mathrm{agreement}_{ap} = 1 - \frac{\left|\mathrm{sim}_{ap}^{(s)} - \mathrm{sim}_{ap}^{(e)}\right|}{2} \tag{19}$$

$$c_{ap} = \mathrm{avg_sim}_{ap} \times \mathrm{agreement}_{ap} \tag{20}$$

$$\tilde{c}_{ap} = \mathrm{clip}\left(\frac{c_{ap}}{\frac{1}{|\mathcal{M}|}\sum_{(u,v)\in\mathcal{M}} c_{uv} + \varepsilon}, c_{\min}, c_{\max} \right). \tag{21}$$

The resulting confidence values are normalized around a mean of 1 and clipped before reweighting the triplet loss. In the current implementation, the clipping range is 0.3–1.2 in the full-batch interface and 0.5–1.5 in the subgraph interface.

$$\mathcal{L}_{\mathrm{tri}} = \frac{1}{|\mathcal{T}|}\sum_{(a,p,n)\in\mathcal{T}} \tilde{c}_{ap} \max\left(\|\mathbf{z}_a - \mathbf{z}_p\|_2 - \|\mathbf{z}_a - \mathbf{z}_n\|_2 + \tau, 0\right). \tag{22}$$

$$p^* = \mathrm{argmin}_{p\in\mathrm{MNN}(a)} \|\mathbf{z}_a - \mathbf{z}_p\|_2 \tag{23}$$

The final objective is the weighted sum of reconstruction loss, relation consistency loss, decoder consistency loss, and confidence-weighted triplet loss. Triplet supervision is activated only in the second training stage:

$$\mathcal{L} = \lambda_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec}} + \lambda_{\mathrm{rc}}\mathcal{L}_{\mathrm{rc}} + \lambda_{\mathrm{dec}}\mathcal{L}_{\mathrm{dec}} + \lambda_{\mathrm{tri}}\mathcal{L}_{\mathrm{tri}} \tag{24}$$

After pretraining, the triplet weight is increased linearly from 0. The maximum triplet weight is exposed as a training parameter and differs between the full-batch and subgraph interfaces. Clipping intervals are heuristic interface-specific safeguards. The full-batch interval permits stronger downweighting while more tightly limiting amplification, whereas the subgraph interval uses a higher lower bound and permits greater amplification. In an audit of 150 saved updates from 30 full-batch runs, weights ranged from 0.874 to 1.007 and clipping was triggered in 0/150 updates; this audit does not validate the subgraph interval (Supplementary Text S1, Section S3 and Table S3).

2.7. Two-Stage Training and Scalable Integration

DVCAlign uses the same dual-view encoder-fusion-decoder architecture for paired and larger multi-slice integration. The difference lies in the training strategy: within-slice representation learning is completed first, and cross-slice alignment is introduced only after the fused embedding has stabilized.

Stage 1 (pretraining). In both the full-batch and subgraph interfaces, the first 500 epochs optimize only the reconstruction loss, relation consistency loss, and decoder consistency loss. This stage establishes the spatial-view, auxiliary-view, and fused representations before MNN supervision is applied.

Stage 2 (alignment training). After pretraining, MNN pairs are recomputed every 100 epochs, including the start of the second stage, so that anchor-positive pairs follow the current alignment state. For larger multi-slice datasets, DVCAlign also provides a subgraph training interface that stores slice-level graphs and performs alignment training on slice subgraphs, reducing memory usage while preserving the same two-stage logic.

2.8. Evaluation and Downstream Domain Identification

Evaluation was designed according to the alignment challenge represented by each benchmark rather than batch mixing alone. The four benchmarks tested laminar preservation in adjacent DLPFC sections, overlap-restricted alignment in partially overlapping mouse brain sections, compartment resolution in heterogeneous breast cancer sections and stage-aware integration in the developmental heart series. When manual annotations were available, integration quality was assessed using clustering agreement together with spatial maps, confusion patterns and marker localization. For the H1 breast cancer comparison, ARI and NMI quantify agreement with the available pathology labels; spatial maps and marker-score contrasts are complementary evidence. For the heart series, where a single shared label set was not appropriate, evaluation focused on stage-resolved spatial maps, embedding visualizations, compositional trends and GO summaries. For the repeated DLPFC comparison, labels from all four sections were pooled before ARI calculation, producing one pooled ARI per execution. Each method was run six times under its documented or default hyperparameters and seed configuration; DVCAlign used training seed 666 and one joint R/mclust EEE fit with K = 7 and clustering seed 666, and run-level values and evaluation metadata were retained for source-data reporting. In the controlled correspondence benchmark, Recall@1 is scored on queries with a true counterpart using all opposite-view spots as candidates. Zero-overlap false acceptance counts unmatched anchors entering MNN triplet supervision, not final registration errors.

After training, the fused embedding was used for spatial-domain identification. For datasets with a prespecified domain number, clustering was performed with mclust [36]. When the domain number was not prespecified, Louvain clustering [37] in Scanpy [34] was used. Integration quality was then assessed according to whether the resulting representation retained spatial domains, marker localization and cross-section organization. Domain labels were treated as analysis outputs rather than predefined biological identities unless supported by annotation or marker evidence.

Supplementary heart GO analyses used one-sided hypergeometric tests with Benjamini–Hochberg correction against stage-specific GO-annotated backgrounds of 9081 and 11,980 genes detected in at least three spots. GO terms contained 10–500 background genes. For the eight-domain rerun, adjustment covered all 24,624 tests, including zero-hit terms, with $q \leq 0.05$ defining significance. Foreground sets followed fixed section-consistent effect-size/detection rules, rather than gene-level FDR selection. Full foreground/background definitions, prespecified domains and separate original-domain/control correction families are documented in Supplementary Text S1.

2.9. Implementation Details

DVCAlign was implemented in Python 3.8. Model training was performed on an NVIDIA GeForce RTX 4090 GPU (NVIDIA, Santa Clara, CA, USA). Unless otherwise stated, the encoder used hidden dimensions of 512 and 32, and the decoder mirrored this architecture through tied weights. The residual combination used fixed weights of 0.7 for the graph-encoded term and 0.3 for the residual term, and the fusion gate was initialized with a spatial prior of 0.7. Models were optimized with Adam [38] in PyTorch 1.9.1 [39] using PyTorch Geometric [40] and gradient clipping at a norm of 5.0. Graph construction

followed the local-density principle described above. Paired or moderate-scale datasets were trained with the full-batch interface, whereas larger multi-slice data were trained with the subgraph interface. An optional chain-consistency term is available in the implementation, but its default weight is 0 and it was not required for the main training setting described here. Runtime and memory summaries for the evaluated datasets are provided in Table 3. In the DLPFC example, settings were 5000 HVGs per section, spatial radius 200, auxiliary neighbors 40/20, learning rate 0.001, 1000 epochs (500 pretraining plus 500 joint-training epochs), MNN k = 20, triplet margin 1.0 and maximum triplet-loss weight 0.7. Training diagnostics from one DLPFC run are reported in Supplementary Text S1 (Table S4). Adam learns model parameters; neighborhood sizes, learning rate and loss coefficients are specified hyperparameters. Training hyperparameters were selected empirically according to the implementation settings, with dataset-specific neighborhood choices and interface-specific training settings as described in Sections 2.3 and 2.6. No separate dataset was held out for hyperparameter validation. Baselines used their documented or default training settings, not equally optimized hyperparameters. The comparisons are therefore conditional on the evaluated configurations, and further tuning could alter the observed performance differences or ranking.

Table 3. Runtime and memory summary for the DVCAlign experiments.

Tissue	# of Slices	Count †	Time	Memory
Human dorsolateral prefrontal cortex (DLPFC)	4	14,364	110 s	6.1 GB
Mouse brain	2	6049	36 s	2.5 GB
HER2-positive human breast cancer	3	1729	26 s	1.4 GB
Developing human embryonic heart	13	1753	58.67 s	6.2 GB

† Heart training used 238 early-stage and 1515 later-stage expression spots in two separate fits. Their training calls, each with 500 pretraining and 500 alignment epochs, took 33.86 s and 24.81 s on a shared NVIDIA RTX 4090, giving 58.67 s when summed. This timing includes auxiliary-graph construction and MNN updates. Data preparation and clustering were handled separately. The coordinate files contain 14,861 registered array positions. Memory values follow the original resource summary.

2.10. Ethics Statement

This study analyzed publicly available datasets and did not involve new human participants or animals; ethical approval was therefore not required.

3. Results

3.1. Benchmark Overview

DVCAlign is designed for structure-preserving integration of spatial transcriptomics (ST) sections when correspondence between slices is not uniform. Adjacent sections may share a repeated anatomical scaffold, whereas other datasets may contain partial overlap, irregular compartments or stage-dependent changes. DVCAlign therefore separates spatial continuity from spatially local molecular similarity and combines the two views before introducing cross-slice supervision.

As shown in Figure 1, DVCAlign takes spatial coordinates and gene-expression profiles from multiple ST sections as input (Figure 1a). It constructs a spatial view and a spatially local expression-similarity view for each slice (Figure 1b), encodes the two graphs in parallel and fuses the resulting embeddings at the spot level (Figure 1c). Cross-slice alignment is then introduced through confidence-weighted supervision, yielding integrated embeddings and downstream spatial domains (Figure 1d).

We evaluated DVCAlign across four benchmarks selected to represent distinct correspondence regimes: repeated laminar anatomy in adjacent DLPFC sections, partial overlap in mouse brain, irregular compartment structure in breast cancer and stage-dependent

organization in the developing heart. Because these regimes pose different evaluation problems, we used complementary evidence rather than a single metric. When reference annotations were available, we used the adjusted Rand index (ARI) together with spatial maps and annotation agreement. For partial-overlap and developmental settings, we further examined domain structure, marker localization, composition and functional summaries.

3.2. DVCAlign Performance on Adjacent DLPFC Section Integration

We first evaluated DVCAlign on four adjacent DLPFC sections (151673-151676) [41], a benchmark in which the major cortical layers recur across slices but local deformation and section-specific variation still make alignment nontrivial (Figure 2). Because manual laminar annotations are available, this dataset provides a relatively controlled test of whether integration improves cross-section correspondence while preserving the ordered cortical architecture from superficial layers to white matter.

Run-to-run variability is a general consideration for spatial transcriptomics integration methods and is not specific to DVCAlign. Stochastic optimization, initialization, matching and clustering can all contribute to this variation, and all methods evaluated here showed some variation across repeated executions. We therefore distinguished representative visualization from aggregate performance evaluation. For the DLPFC comparison, labels from all four sections were concatenated before ARI calculation, yielding one pooled ARI per execution. Each method was evaluated under its documented or default hyperparameters, and all six prespecified executions were reported per method without score-based selection. The same preprocessing, clustering and pooled-ARI calculation were applied to all methods. Each method retained its own default seed configuration across these executions; DVCAlign used seed 666. These are computational repetitions of four sections from one donor, not independent biological replicates.

Across the six executions, the mean pooled ARI (sample s.d.) was 0.633 (0.007) for DVCAlign, 0.611 (0.024) for STAligner, 0.605 (0.031) for MaskGraphene, 0.582 (0.042) for SpaBatch, 0.572 (0.023) for SEDR, 0.382 (0.059) for PASTE and 0.278 (0.053) for Harmony (Figure 2d). DVCAlign therefore achieved the highest mean pooled ARI under the evaluated default-setting protocol. The maps in Figure 2a–c show one representative execution used for visualization; the ARI displayed in each method header is the pooled ARI for that execution and is not the across-execution mean. The run-level distribution in Figure 2d, rather than the single visualization run, provides the quantitative comparison. The original six-run values, conditional uncertainty and 18 additional executions using training seeds 0, 1 and 42 are reported in Supplementary Text S1 (Tables S5–S8). The difference in reported means does not establish statistical superiority or between-donor generalization.

The spatial maps provide visual context for the quantitative comparison (Figure 2a–c). PASTE preserved the broad laminar scaffold but showed boundary displacement, particularly in the lower regions of the sections. MaskGraphene recovered the global layer order, although several intermediate bands appeared smoother or less sharply separated. DVCAlign maintained the shared laminar progression across all four slices while retaining clearer transitions between neighboring layers. This pattern was consistent with improved recovery of layer-resolved cortical structure rather than simple smoothing between adjacent layers. Additional method comparisons are shown in Figure S3a.

Additional analyses provided complementary support for this interpretation. Agreement between manual cortical-layer annotations and mclust domains was concentrated near the diagonal, and predicted domain proportions were highly correlated across sections ($r = 0.94$–0.99; Figure S3b,c). GFAP remained enriched in the white-matter-associated region across sections (Figure 2e), while a Uniform Manifold Approximation and Projection (UMAP) visualization of the integrated embeddings showed mixing by slice alongside

ordered ground-truth layers and inferred domains (Figure 2f). Together, these results support DVCAlign as a method for improving adjacent-section embedding integration while retaining anatomically meaningful laminar structure. A complementary DLPFC target-truncation comparison (source 151673, target 151674) showed higher label-transfer ARI for DVCAlign than PASTE2 across five retention levels on common target spots receiving PASTE2 correspondences (Table S9).

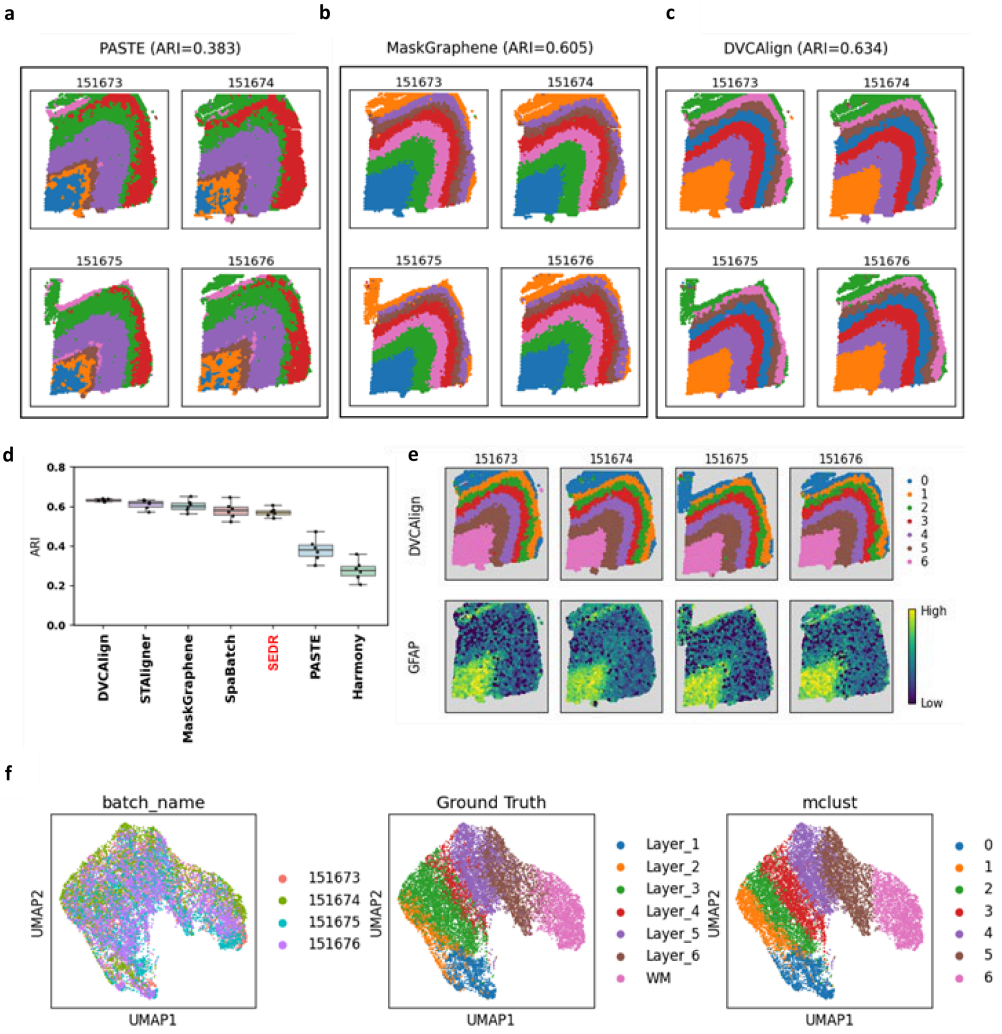

Figure 2. DVCAlign preserves coherent laminar organization in adjacent DLPFC sections. Run-to-run variability is a general consideration for spatial transcriptomics integration methods and is not specific to DVCAlign; all methods evaluated here showed some variation across repeated executions. Accordingly, panels (**a**–**c**) show representative single executions for visualization, whereas panel (**d**) summarizes six repeated executions for quantitative comparison. (**a**–**c**), Spatial-domain maps from one representative execution for PASTE (**a**), MaskGraphene (**b**) and DVCAlign (**c**) across sections 151673-151676. The ARI in each method header is the pooled ARI for that individual execution and is not the across-execution mean. (**d**) Distribution of pooled ARI values over six executions per method under the documented or default evaluation settings. Colors denote inferred spatial domains in (**a**–**c**) and distinguish methods in (**d**). Black dots represent individual executions, boxes indicate the interquartile range, center lines indicate medians and whiskers represent 1.5 times the interquartile range. (**e**) DVCAlign spatial domains with representative GFAP maps across the four sections. (**f**) UMAP embeddings colored by slice, ground truth and inferred domains. Supplementary method comparisons and agreement analyses are shown in Figure S3.

3.3. DVCAlign Performance on Partially Overlapping Mouse Brain Sections

We next evaluated DVCAlign on partially overlapping mouse brain sagittal sections, where only part of the tissue should be aligned and large unmatched regions must remain distinct (Figure 3). This setting is more demanding than adjacent DLPFC integration because successful alignment requires selectivity: a good method should recover the true shared interface without propagating false correspondence into non-overlapping territories.

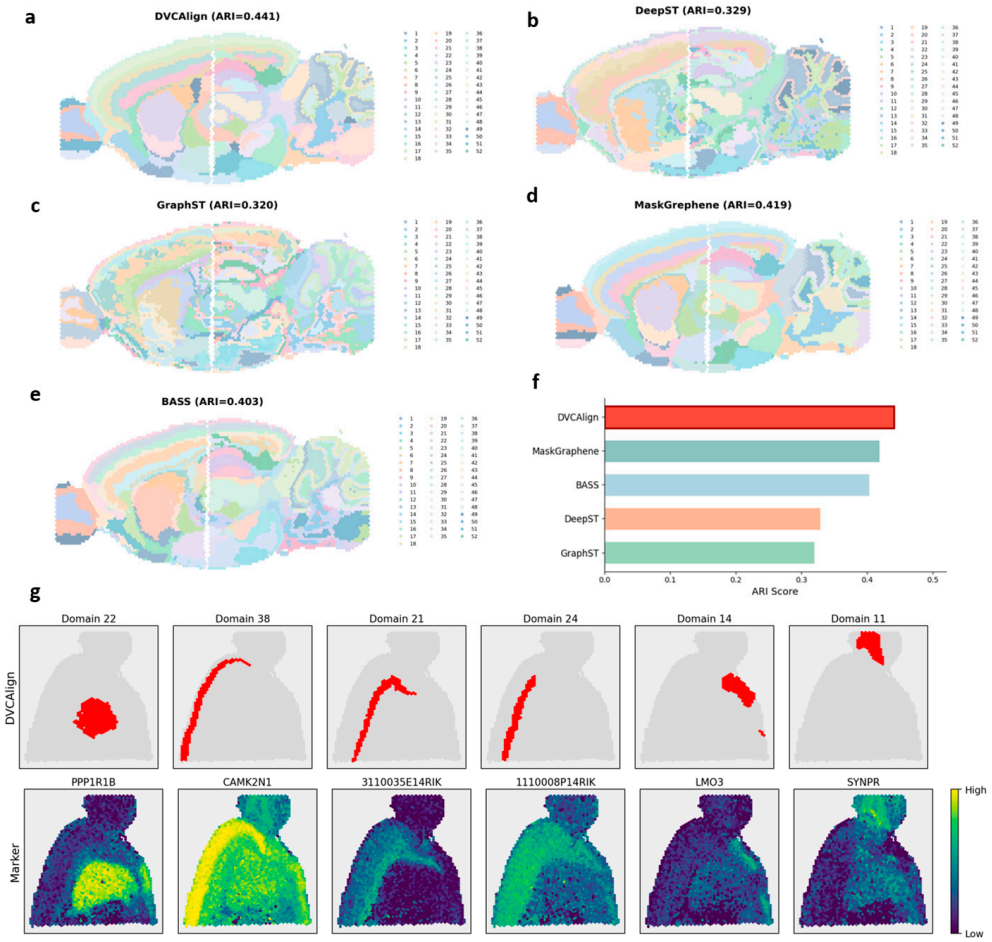

Figure 3. DVCAlign preserves localized structure in partially overlapping mouse brain sagittal sections. (**a**) DVCAlign domain map. (**b**) DeepST domain map. (**c**) GraphST domain map. (**d**) MaskGraphene domain map. (**e**) BASS domain map. The numbers in the legends of (**a**–**e**) identify the inferred domains for each method. (**f**) ARI comparison across the five methods shown. (**g**) Representative DVCAlign domains and matched marker-expression patterns. Additional spatial marker-expression patterns and the domain-associated marker dot plot are shown in Figure S4a and Figure S4b, respectively.

Because this benchmark still provides a reference structure, we first compared methods quantitatively. DVCAlign achieved an ARI of 0.441, compared with 0.419 for MaskGraphene, 0.403 for BASS [42], 0.329 for DeepST [43], and 0.320 for GraphST (Figure 3f). DVCAlign therefore achieved the highest ARI among the five methods shown in Figure 3f.

The spatial maps show where the difference was most apparent (Figure 3a–c). DeepST and GraphST produced more diffuse or fragmented partitions, whereas DVCAlign preserved a compact shared interface and clearer separation between the two non-overlapping sides of the tissue. This visual pattern was consistent with the ARI comparison in Figure 3f. In this benchmark, continuity outside the shared region would indicate over-alignment rather than successful recovery of the true overlap. These qualitative maps do not establish exact spot correspondence or hard rejection of unmatched anchors.

The downstream analyses provided additional support. The domain-associated marker dot plot retained structured expression patterns across recovered domains (Figure S4b). The representative DVCAlign domains in Figure 3g were accompanied by spatially localized marker-expression patterns, including PPP1R1B, CAMK2N1, 3110035E14Rik, 1110008P14Rik, LMO3 and SYNPR. Supplementary spatial expression maps for COL1A2, IGFBP2, MGP, NNAT, NOV and PTGDS are shown in Figure S4a. Together, these expression-derived patterns provide qualitative corroboration of domain structure. Controlled MA/MP technical-copy pairs at 100%, 75%, 50%, 25% and 0% overlap provided direct correspondence tests (Figure S5). At 50% overlap, DVCAlign improved Recall@1 over matched STAligner by 4.48–6.50 percentage points. At zero overlap, 29.39–48.53% of unmatched anchors still entered mutual nearest-neighbor (MNN) supervision; this is supervision-acceptance error, not final registration error. The controlled copies do not independently establish anatomical correspondence in the real MA–MP pair.

3.4. DVCAlign Performance on Heterogeneous Human Breast Cancer Sections

We then tested DVCAlign on three heterogeneous human breast cancer sections (H1–H3), a setting in which local compartment boundaries are irregular and nearby regions can remain spatially close while carrying distinct molecular programs (Figure 4). Unlike the repeated laminar structure in DLPFC or the interface-restricted geometry of the mouse brain benchmark, this dataset asks whether an integration method can preserve complex compartmental organization without oversmoothing local transitions.

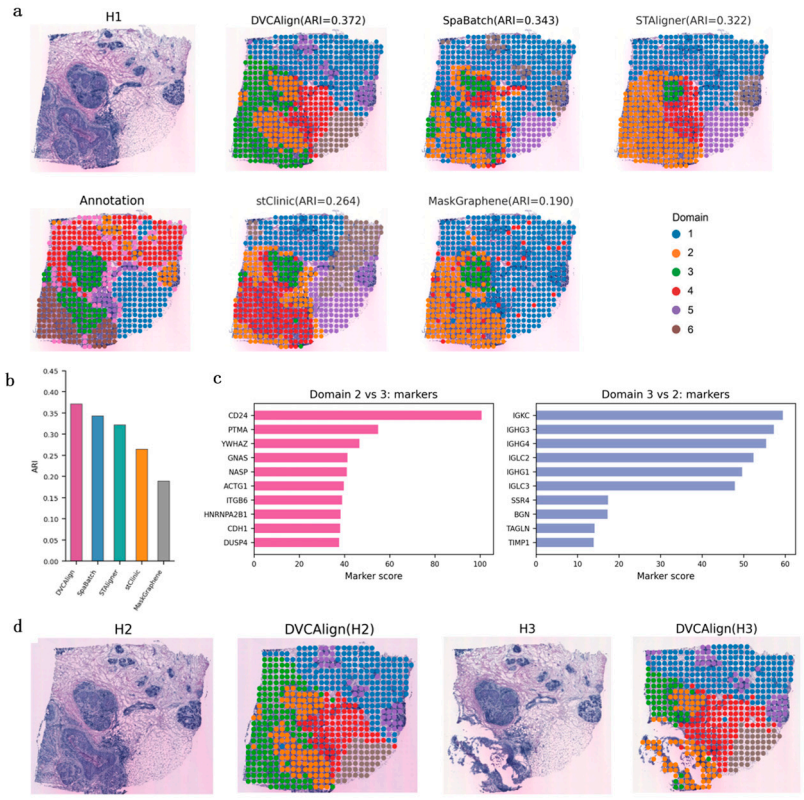

Figure 4. Spatial and molecular assessment of compartment structure across breast cancer sections. (**a**) H1 histology, manual annotation and method-specific domain maps for DVCAlign, SpaBatch, STAligner, stClinic and MaskGraphene. (**b**) H1 ARI comparison for DVCAlign and the four comparators; reference annotations are used only for evaluation. (**c**) Marker-score contrasts between Domains 2 and 3 in H1; all 11 genes were included in the integration features. (**d**) Histology and DVCAlign-predicted domains for H2 and H3.

H1 provided the most direct visual comparison because both histology and manual annotation were available. We compared method-specific domain maps with the annotated compartment layout (Figure 4a). Quantitatively, DVCAlign achieved an ARI of 0.372, compared with 0.343 for SpaBatch, 0.322 for STAligner, 0.264 for stClinic and 0.190 for MaskGraphene (Figure 4b). Using the same saved clustering, agreement with H&E-based pathology was evaluated using ARI and normalized mutual information (NMI), yielding ARI/NMI = 0.372/0.494 on 613 annotated spots including undetermined labels, and 0.471/0.572 on 530 spots in six defined categories excluding undetermined labels (Supplementary Text S1, Table S10). These are different scoring scopes, not refitted clusterings. Because unsupervised domain labels need not correspond one-to-one to annotation labels, we treated the annotation as a spatial reference rather than as a direct assignment of biological identity. Spatial maps, marker-score contrasts and cross-section consistency provide complementary evidence.

The H1 maps showed that all methods recovered parts of the broad tissue organization, but differed in the placement and sharpness of local boundaries (Figure 4a). DVCAlign reproduced several prominent spatially contiguous regions, including a large central region and a lower-right compartment, while retaining transitions between neighboring regions. SpaBatch and STAligner captured parts of the broad arrangement but showed more local mixing or merging in some areas. stClinic and MaskGraphene produced more coarsely separated patterns in several central and peripheral areas. These visual differences support preservation of a plausible compartmental organization in H1, but they do not by themselves establish complete agreement with the manual annotation or a definitive ranking across methods.

Marker-score contrasts provided molecular support for two of the recovered H1 domains (Figure 4c). Domain 2 was characterized by CD24, PTMA, YWHAZ, CDH1 and ITGB6, whereas Domain 3 was characterized by immunoglobulin-related genes including IGKC, IGHG3, IGHG4, IGLC2, IGHG1 and IGLC3. This contrast is consistent with an epithelial-like or tumor-associated program in Domain 2 and an immunoglobulin-rich program in Domain 3, but these domain labels should not be interpreted as definitive cell-type identities without additional validation. In H2 and H3, the DVCAlign maps retained a broadly comparable arrangement of several large compartments despite changes in tissue shape and local morphology (Figure 4d). Thus, Figure 4 supports preservation of spatially organized and transcriptionally distinct compartments, while the extent of agreement varies across sections. All 11 highlighted markers were included in the integration features, so their expression patterns are corroborative rather than independent molecular validation. H2/H3 assessment remains qualitative because corresponding independent pathology annotations are unavailable.

3.5. DVCAlign Performance on a Developmental Human Embryonic Heart Series

Finally, we applied DVCAlign to a developmental human embryonic heart series spanning 4.5-5 and 6.5 post-conception weeks (PCW) [44] (Figure 5). This benchmark differs from the previous three because the slices are not expected to collapse to one repeated structure. Instead, the key question is whether integration can recover recurrent anatomical organization within each developmental stage while preserving stage-associated differences. Because a single shared label set is not appropriate here, evaluation must rely on spatial organization, latent structure, compositional trends, and functional interpretation rather than annotation agreement alone. The four early-stage and nine later-stage sections represent one heart specimen per stage; stage is therefore confounded with specimen and technical-batch factors. The stages used the same architecture but separate fitted models, with K = 2 and K = 6 prespecified following the stage-wise application in [24].

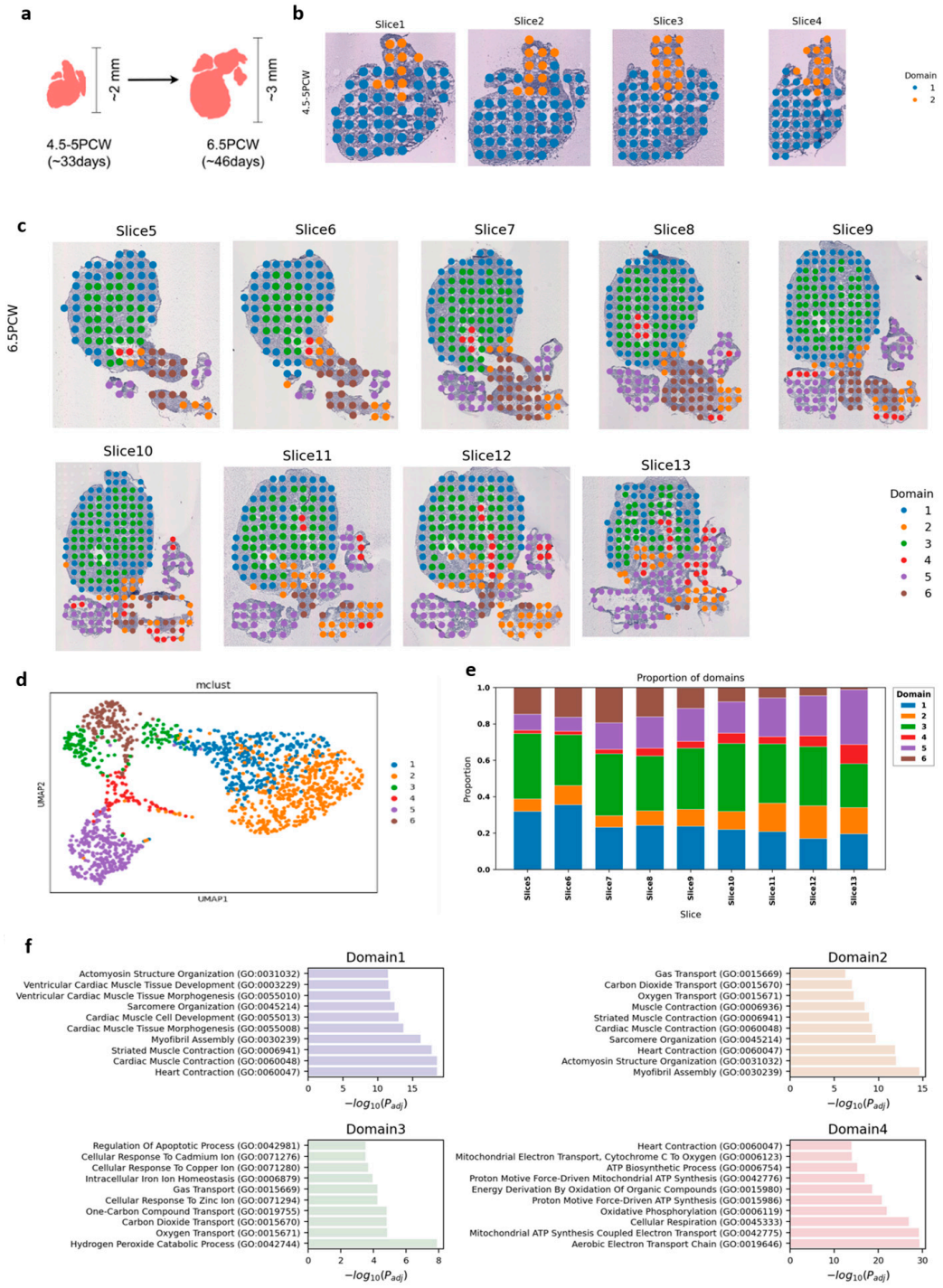

Figure 5. DVCAlign describes stage-associated organization across a developing heart series. (**a**) Developmental schematic from 4.5-5 PCW to 6.5 PCW. (**b**) Spatial-domain maps across four 4.5-5 PCW slices. (**c**) Spatial-domain maps across nine 6.5 PCW slices. (**d**) UMAP visualization of the resulting stage-wise embeddings. (**e**) Slice-wise domain proportions for the 6.5 PCW slices. (**f**) Representative domain-associated GO summaries. The stages used separate fitted models with prespecified K = 2 and K = 6 and represent one specimen per stage. Analyses used 238 early-stage and 1515 later-stage expression spots (1753 total); the 14,861 registered array positions in Table 1 are a different count.

The spatial maps showed greater structural complexity at the later developmental stage. The four 4.5-5 PCW slices were dominated by a simpler two-domain organization (Figure 5b), whereas the 6.5 PCW slices contained a richer six-domain structure with additional peripheral and internal regions (Figure 5c). Because K was prespecified and the

stages were fitted separately, this two-versus-six-domain pattern is descriptive and does not demonstrate separation of developmental effects from batch effects.

The latent embedding visualization and compositional summary described this organization. In the UMAP visualization, the recovered domains occupied structured regions rather than collapsing into one mixed cluster (Figure 5d). Domain proportions remained variable across the 6.5 PCW slices, with Domain 5 particularly prominent in Slice13 (Figure 5e). These results describe recurrent organization and later-stage local heterogeneity without independently validating spot-level correspondence.

Gene Ontology (GO) enrichment provided descriptive annotation for the recovered domains. Domains 1 and 2 were enriched for cardiac muscle and contractile programs, including ventricular cardiac muscle development, sarcomere organization, myofibril assembly, striated muscle contraction and heart contraction (Figure 5f). Domain 3 was associated with stress- and ion-response processes, whereas Domain 4 showed exploratory functional summaries for oxidative phosphorylation, ATP biosynthesis and mitochondrial respiratory pathways. These expression-derived patterns are corroborative rather than independent molecular validation. In a supplementary stage-wise rerun, cardiac muscle contraction was enriched with 11/163 GO-eligible foreground genes, a fold enrichment of 24.50 and global $q = 7.05 \times 10^{-10}$. Oxidative phosphorylation, ATP biosynthesis and the queried mitochondrial ATP-synthesis term did not reach the global significance threshold. Rerun region labels are not assumed to match the original Figure 5f domain labels (Supplementary Text S1, Table S11).

Original Figure 5 domains D5 (279 spots) and D6 (153 spots) were further annotated using marker expression and GO evidence. D5 was associated with atrial-myocardium programs (MYH6/MYL7), whereas D6 was associated with outflow-tract/large-vessel programs (ELN/MYH11/ACTA2). The within-versus-outside expression contrasts and multiple-testing-adjusted GO statistics are provided in Supplementary Text S1 (Table S12). These are expression-based functional associations, not independent molecular validation or evidence of cell lineage. Shared-encoder and matched-K controls retained contraction- and matrix-associated themes (Table S13).

3.6. Cross-Benchmark Summary of Structure-Preserving Integration

Taken together, the four benchmarks tested whether DVCAlign could handle four different correspondence assumptions: repeated anatomy, partial overlap, irregular compartment structure and stage-associated variation. The first setting emphasized laminar preservation, the second required restriction of alignment to a shared interface, the third required retention of local tumor-associated compartments and the fourth required shared organization to be recovered with stage-associated patterns interpreted under specimen and batch confounding.

Across these settings, DVCAlign showed a consistent evaluation pattern. It achieved the highest mean pooled ARI in the original six-execution DLPFC comparison and the highest ARI among the evaluated mouse brain methods. The breast cancer analysis included H1 pathology agreement, spatial maps and marker localization, whereas the heart analysis emphasized spatial organization, composition and functional annotation. The combined evidence therefore supports embedding integration under the evaluated conditions, but the strength of each conclusion depends on the evidence available for that benchmark. Targeted ablations of the graph views, adaptive fusion, confidence weighting, triplet objective and staged optimization reduced ARI relative to the complete model across H1, DLPFC and mouse MA under the reported settings (Figure S1). These component effects are setting-dependent and do not establish universal necessity.

4. Discussion

Across four distinct alignment settings, the results support DVCAlign as an embedding integration framework under the evaluated conditions. It improved clustering agreement and retained structured spatial patterns, although unmatched-anchor errors and specimen-confounded stage effects remained possible. This interpretation is based on complementary evidence, including repeated ARI values, spatial-domain maps, marker localization, composition summaries and functional annotation.

This interpretation is consistent with the design of DVCAlign. The framework separates spatial continuity from spatially local molecular similarity, fuses the two views at the spot level and introduces cross-slice alignment after within-slice representation learning. These design choices provide a plausible explanation for the observed behavior, and are additionally evaluated by the targeted component ablations in Figure S1. These comparisons support measurable contributions under the tested settings without establishing universal optimality.

The present results also highlight a broader principle for ST integration. Integration should not be judged by batch mixing alone, because apparent mixing can improve even when anatomical boundaries, compartment identity or developmental ordering have been weakened. In this study, interpretability was assessed through coherent spatial maps, localized marker patterns, agreement with external annotations where available, preserved compositional trends and functionally annotated domains. These criteria provide a more informative evaluation of structure-preserving integration than mixing alone.

Despite these strengths, DVCAlign has several limitations. Its performance depends on recoverable within-slice neighborhood structure and cross-slice relationships strong enough to support plausible mutual nearest-neighbor correspondences. Extremely sparse tissue, highly discontinuous sections, minimal biological overlap or insufficient shared features may therefore remain challenging. The current framework also does not explicitly model uncertainty in cross-slice matching. The targeted ablations do not establish generalization across all ST platforms or tissues. The DLPFC repetitions are computational executions from one donor, and the heart comparison has one specimen per stage; neither provides independent biological replication. Future work should address uncertainty-aware correspondence modeling, broader component validation and broader cross-platform validation. Within these bounds, the present results support DVCAlign for the evaluated spatial alignment regimes. Confidence weights are relative consistency multipliers rather than calibrated correctness probabilities. At zero overlap, the mean proportion of unmatched anchors entering MNN triplet supervision was 29.39–48.53%, underscoring the importance of shared biological overlap for reliable correspondence; feedback diagnostics also identified persistence of incorrect high-confidence pairs (Figures S2 and S5). Visual correspondence alone cannot rule out local over-alignment. Cell-type deconvolution could complement interpretation of the original count data, but its effect on integration performance was not evaluated.

5. Conclusions

DVCAlign combines spatially constrained molecular neighborhoods, adaptive spot-wise fusion and consistency-weighted cross-slice supervision for embedding integration. In the evaluated default-setting benchmarks, it improved clustering agreement, and targeted component ablations supported measurable contributions. Controlled correspondence and confidence diagnostics also revealed unmatched-anchor acceptance and residual feedback risk. These findings support condition-dependent spatial-domain analysis, not guaranteed anatomical registration. Broader donor and platform validation, calibrated correspondence uncertainty and independent biological validation remain necessary.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/biology15232210/s1.

Author Contributions: C.M. and W.C. contributed equally to this work. Conceptualization, C.M., W.C. and Y.J.; methodology, C.M. and W.C.; software, formal analysis and visualization, W.C.; data curation and validation, X.A. and Y.X.; writing—original draft preparation, C.M. and W.C.; writing—review and editing, all authors; supervision and project administration, Y.J.; funding acquisition, C.M. and Y.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the China Postdoctoral Science Foundation (grant number 2025MD784116) and the Chongqing Science and Technology Commission (grant number CSTB2025NSCQ-LZX0077).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All primary datasets analyzed in this study are available through the sources and access procedures listed in Table 2: spatialLIBD for human DLPFC sections 151673-151676; the 10x Genomics Mouse Brain Serial Section 2 records for sagittal-anterior and sagittal-posterior sections; the HER2ST repository and Zenodo record 4751624 for the breast cancer sections; and the human embryonic heart records at Mendeley Data (doi: 10.17632/dgnysc3zn5.1) and EGA (EGAD00001005468). Access to the EGA sequencing record is controlled and subject to authorization by the relevant Data Access Committee; this is distinct from publicly available processed resources. The full DVCAlign implementation, installation instructions and dependency files are publicly available at https://github.com/weic74285-pixel/DVCAlign (accessed on 19 November 2026). The repository includes runnable examples for human DLPFC (examples/DLPFC.ipynb) and partially overlapping mouse brain data (examples/MB2SAP.ipynb). These notebooks demonstrate data preparation, spatial-graph construction, model training, clustering and evaluation. Dataset access sources are listed in Table 2 and the repository README.

Acknowledgments: We thank the teams who generated and publicly released the spatial transcriptomics datasets used in this study.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Method of the Year 2020: Spatially resolved transcriptomics. *Nat. Methods* **2021**, *18*, 1. [CrossRef] [PubMed]
2. Moses, L.; Pachter, L. Museum of spatial transcriptomics. *Nat. Methods* **2022**, *19*, 534–546. [CrossRef] [PubMed]
3. Ståhl, P.L.; Salmén, F.; Vickovic, S.; Lundmark, A.; Fernández Navarro, J.; Magnusson, J.; Giacomello, S.; Asp, M.; Westholm, J.O.; Huss, M.; et al. Visualization and analysis of gene expression in tissue sections by spatial transcriptomics. *Science* **2016**, *353*, 78–82. [CrossRef] [PubMed]
4. Rodriques, S.G.; Stickels, R.R.; Goeva, A.; Martin, C.A.; Murray, E.; Vanderburg, C.R.; Welch, J.; Chen, L.M.; Chen, F.; Macosko, E.Z. Slide-seq: A scalable technology for measuring genome-wide expression at high spatial resolution. *Science* **2019**, *363*, 1463–1467. [CrossRef] [PubMed]
5. Chen, K.H.; Boettiger, A.N.; Moffitt, J.R.; Wang, S.; Zhuang, X. Spatially resolved, highly multiplexed RNA profiling in single cells. *Science* **2015**, *348*, aaa6090. [CrossRef] [PubMed]
6. Wang, X.; Allen, W.E.; Wright, M.A.; Sylwestrak, E.L.; Samusik, N.; Vesuna, S.; Evans, K.; Liu, C.; Ramakrishnan, C.; Liu, J.; et al. Three-dimensional intact-tissue sequencing of single-cell transcriptional states. *Science* **2018**, *361*, eaat5691. [CrossRef] [PubMed]
7. Chen, A.; Liao, S.; Cheng, M.; Ma, K.; Wu, L.; Lai, Y.; Qiu, X.; Yang, J.; Xu, J.; Hao, S.; et al. Spatiotemporal transcriptomic atlas of mouse organogenesis using DNA nanoball-patterned arrays. *Cell* **2022**, *185*, 1777–1792. [CrossRef] [PubMed]
8. Rao, A.; Barkley, D.; França, G.S.; Yanai, I. Exploring tissue architecture using spatial transcriptomics. *Nature* **2021**, *596*, 211–220. [CrossRef] [PubMed]
9. Zhuang, X. Spatially resolved single-cell genomics and transcriptomics by imaging. *Nat. Methods* **2021**, *18*, 18–22. [CrossRef] [PubMed]
10. Moffitt, J.R.; Lundberg, E.; Heyn, H. The emerging landscape of spatial profiling technologies. *Nat. Rev. Genet.* **2022**, *23*, 741–759. [CrossRef] [PubMed]

11. Longo, S.K.; Guo, M.G.; Ji, A.L.; Khavari, P.A. Integrating single-cell and spatial transcriptomics to elucidate intercellular tissue dynamics. *Nat. Rev. Genet.* **2021**, *22*, 627–644. [CrossRef] [PubMed]
12. Luecken, M.D.; Theis, F.J. Current best practices in single-cell RNA-seq analysis: A tutorial. *Mol. Syst. Biol.* **2019**, *15*, e8746. [CrossRef] [PubMed]
13. Korsunsky, I.; Millard, N.; Fan, J.; Slowikowski, K.; Zhang, F.; Wei, K.; Baglaenko, Y.; Brenner, M.; Loh, P.-R.; Raychaudhuri, S. Fast, sensitive and accurate integration of single-cell data with Harmony. *Nat. Methods* **2019**, *16*, 1289–1296. [CrossRef] [PubMed]
14. Stuart, T.; Butler, A.; Hoffman, P.; Hafemeister, C.; Papalexi, E.; Mauck, W.M.; Hao, Y.; Stoeckius, M.; Smibert, P.; Satija, R. Comprehensive Integration of Single-Cell Data. *Cell* **2019**, *177*, 1888–1902.e21. [CrossRef] [PubMed]
15. Hie, B.; Bryson, B.; Berger, B. Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. *Nat. Biotechnol.* **2019**, *37*, 685–691. [CrossRef] [PubMed]
16. Polański, K.; Young, M.D.; Miao, Z.; Meyer, K.B.; Teichmann, S.A.; Park, J.E. BBKNN: Fast batch alignment of single cell transcriptomes. *Bioinformatics* **2020**, *36*, 964–965. [CrossRef] [PubMed]
17. Dong, K.; Zhang, S. Deciphering spatial domains from spatially resolved transcriptomics with an adaptive graph attention auto-encoder. *Nat. Commun.* **2022**, *13*, 1739. [CrossRef] [PubMed]
18. Hu, J.; Li, X.; Coleman, K.; Schroeder, A.; Ma, N.; Irwin, D.J.; Lee, E.B.; Shinohara, R.T.; Li, M. SpaGCN: Integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. *Nat. Methods* **2021**, *18*, 1342–1351. [CrossRef] [PubMed]
19. Long, Y.; Ang, K.S.; Li, M.; Chong, K.L.K.; Sethi, R.; Zhong, C.; Xu, H.; Ong, Z.; Sachaphibulkij, K.; Chen, A.; et al. Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST. *Nat. Commun.* **2023**, *14*, 1155. [CrossRef] [PubMed]
20. Xu, H.; Fu, H.; Long, Y.; Ang, K.S.; Sethi, R.; Chong, K.; Li, M.; Uddamvathanak, R.; Lee, H.K.; Ling, J.; et al. Unsupervised spatially embedded deep representation of spatial transcriptomics. *Genome Med.* **2024**, *16*, 12. [CrossRef] [PubMed]
21. Zeira, R.; Land, M.; Strzalkowski, A.; Raphael, B.J. Alignment and integration of spatial transcriptomics data. *Nat. Methods* **2022**, *19*, 567–575. [CrossRef] [PubMed]
22. Liu, X.; Zeira, R.; Raphael, B.J. Partial alignment of multislice spatially resolved transcriptomics data. *Genome Res.* **2023**, *33*, 1124–1132. [CrossRef] [PubMed]
23. Zhou, X.; Dong, K.; Zhang, S. Integrating spatial transcriptomics data across different conditions, technologies, and developmental stages. *Nat. Comput. Sci.* **2023**, *3*, 894–906. [CrossRef] [PubMed]
24. Niu, J.; Fang, D.; Chen, J.; Xiong, Y.; Liu, J.; Min, W. SpaBatch: Deep Learning-Based Cross-Slice Integration and 3D Spatial Domain Identification in Spatial Transcriptomics. *Adv. Sci.* **2025**, *12*, e09090. [CrossRef] [PubMed]
25. Hu, Y.; Lin, Z.; Xie, M.; Yuan, W.; Li, Y.; Rao, M.; Liu, Y.H.; Shen, W.; Zhang, L.; Zhou, X.M. MaskGraphene: An advanced framework for interpretable joint representation for multi-slice, multi-condition spatial transcriptomics. *Genome Biol.* **2025**, *26*, 380. [CrossRef] [PubMed]
26. Zuo, C.; Xia, J.; Xu, Y.; Xu, Y.; Gao, P.; Zhang, J.; Wang, Y.; Chen, L. stClinic: Dissects clinically relevant niches by integrating spatial multi-slice multi-omics data in dynamic graphs. *Nat. Commun.* **2025**, *16*, 5317. [CrossRef] [PubMed]
27. Liu, W.; Liao, X.; Luo, Z.; Yang, Y.; Lau, M.C.; Jiao, Y.; Shi, X.; Zhai, W.; Ji, H.; Yeong, J.; et al. Probabilistic embedding, clustering, and alignment for integrating spatial transcriptomics data with PRECAST. *Nat. Commun.* **2023**, *14*, 296. [CrossRef] [PubMed]
28. Clifton, K.; Anant, M.; Aihara, G.; Atta, L.; Aimiuwu, O.K.; Kebschull, J.M.; Miller, M.I.; Tward, D.; Fan, J. STalign: Alignment of spatial transcriptomics data using diffeomorphic metric mapping. *Nat. Commun.* **2023**, *14*, 8123. [CrossRef] [PubMed]
29. Xia, C.R.; Cao, Z.J.; Tu, X.M.; Gao, G. Spatial-linked alignment tool (SLAT) for aligning heterogenous slices. *Nat. Commun.* **2023**, *14*, 7236. [CrossRef] [PubMed]
30. Tang, Z.; Luo, S.; Zeng, H.; Huang, J.; Sui, X.; Wu, M.; Wang, X. Search and match across spatial omics samples at single-cell resolution. *Nat. Methods* **2024**, *21*, 1818–1829. [CrossRef] [PubMed]
31. Klein, D.; Palla, G.; Lange, M.; Klein, M.; Piran, Z.; Gander, M.; Meng-Papaxanthos, L.; Sterr, M.; Saber, L.; Jing, C.; et al. Mapping cells through time and space with moscot. *Nature* **2025**, *638*, 1065–1075. [CrossRef] [PubMed]
32. Jones, A.; Townes, F.W.; Li, D.; Engelhardt, B.E. Alignment of spatial genomics data using deep Gaussian processes. *Nat. Methods* **2023**, *20*, 1379–1387. [CrossRef] [PubMed]
33. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph attention networks. *arXiv* **2018**, arXiv:1710.10903. Available online: https://arxiv.org/abs/1710.10903 (accessed on 19 November 2026).
34. Wolf, F.A.; Angerer, P.; Theis, F.J. Scanpy: Large-scale single-cell gene expression data analysis. *Genome Biol.* **2018**, *19*, 15. [CrossRef] [PubMed]
35. Haghverdi, L.; Lun, A.T.L.; Morgan, M.D.; Marioni, J.C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat. Biotechnol.* **2018**, *36*, 421–427. [CrossRef] [PubMed]
36. Scrucca, L.; Fop, M.; Murphy, T.B.; Raftery, A.E. Mclust 5: Clustering, classification and density estimation using Gaussian finite mixture models. *R J.* **2016**, *8*, 289–317. [CrossRef]

37. Blondel, V.D.; Guillaume, J.L.; Lambiotte, R.; Lefebvre, E. Fast unfolding of communities in large networks. *J. Stat. Mech. Theory Exp.* **2008**, *2008*, P10008. [CrossRef]
38. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2015**, arXiv:1412.6980. Available online: https://arxiv.org/abs/1412.6980 (accessed on 19 November 2026).
39. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 8024–8035. Available online: https://arxiv.org/abs/1912.01703 (accessed on 19 November 2026).
40. Fey, M.; Lenssen, J.E. Fast graph representation learning with PyTorch Geometric. *arXiv* **2019**, arXiv:1903.02428. Available online: https://arxiv.org/abs/1903.02428 (accessed on 19 November 2026).
41. Maynard, K.R.; Collado-Torres, L.; Weber, L.M.; Uytingco, C.; Barry, B.K.; Williams, S.R.; Catallini, J.L.; Tran, M.N.; Besich, Z.; Tippani, M.; et al. Transcriptome-scale spatial gene expression in the human dorsolateral prefrontal cortex. *Nat. Neurosci.* **2021**, *24*, 425–436. [CrossRef] [PubMed]
42. Li, Z.; Zhou, X. BASS: Multi-scale and multi-sample analysis enables accurate cell type clustering and spatial domain detection in spatial transcriptomic studies. *Genome Biol.* **2022**, *23*, 168. [CrossRef] [PubMed]
43. Xu, C.; Jin, X.; Wei, S.; Wang, P.; Luo, M.; Xu, Z.; Yang, W.; Cai, Y.; Xiao, L.; Lin, X.; et al. DeepST: Identifying spatial domains in spatial transcriptomics by deep learning. *Nucleic Acids Res.* **2022**, *50*, e131. [CrossRef] [PubMed]
44. Asp, M.; Giacomello, S.; Larsson, L.; Wu, C.; Fürth, D.; Qian, X.; Wärdell, E.; Custodio, J.; Reimegård, J.; Salmén, F.; et al. A spatiotemporal organ-wide gene expression and cell atlas of the developing human heart. *Cell* **2019**, *179*, 1647–1660. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.