Review article

A comprehensive overview of graph neural network-based approaches to clustering for spatial transcriptomics

Teng Liu [a,b,1,2], Zhao-Yu Fang [c,1,3], Zongbo Zhang [a,4], Yongxiang Yu [a,5], Min Li [c,d,*,6], Ming-Zhu Yin [a,b,**,7]

[a] Clinical Research Center (CRC), Clinical Pathology Center (CPC), Cancer Early Detection and Treatment Center (CEDTC) and Translational Medicine Research Center (TMRC), Chongqing University Three Gorges Hospital, Chongqing University, Wanzhou, Chongqing, China
[b] Chongqing Technical Innovation Center for Quality Evaluation and Identification of Authentic Medicinal Herbs, Chongqing, China
[c] School of Computer Science and Engineering, Central South University, Changsha 410083, China
[d] Hunan Provincial Engineering Research Center of Intelligent Computing in Biology and Medicine, Central South University, Changsha 410083, China

ARTICLE INFO

ABSTRACT

Spatial transcriptomics technologies enable researchers to accurately quantify and localize messenger ribonucleic acid (mRNA) transcripts at a high resolution while preserving their spatial context. The identification of spatial domains, or the task of spatial clustering, plays a crucial role in investigating data on spatial transcriptomes. One promising approach for classifying spatial domains involves the use of graph neural networks (GNNs) by leveraging gene expressions, spatial locations, and histological images. This study provided a comprehensive overview of the most recent GNN-based methods of spatial clustering methods for the analysis of data on spatial transcriptomics. We extensively evaluated the performance of current methods on prevalent datasets of spatial transcriptomics by considering their accuracy of clustering, robustness, data stabilization, relevant requirements, computational efficiency, and memory use. To this end, we explored 60 clustering scenarios by extending the essential frameworks of spatial clustering for the selection of the GNNs, algorithms of downstream clustering, principal component analysis (PCA)-based reduction, and refined methods of correction. We comparatively analyzed the performance of the methods in terms of spatial clustering to identify their limitations and outline future directions of research in the field. Our survey yielded novel insights, and provided motivation for further investigating spatial transcriptomics.

1. Introduction

Spatial transcriptomics technologies have facilitated the profiling of genome-wide readouts and the documentation of the spatial locations of individual cells [1]. This wealth of information on gene expressions and their spatial contexts has enabled researchers to identify cancer initiation and disease progression. Furthermore, researchers can gain insights into the microenvironments of tissues and cell-to-cell communications by leveraging information on their spatial positions [2]. Technologies of spatial transcriptomics can be broadly categorized into imaging-based and sequencing-based techniques. Imaging-based methods include MERFISH [3], seqFISH [4], and osmFISH [5], while sequencing-based methods include 10x Visium [6], Slide-seqV2 [7], and Stereo-seq [8]. Each of these techniques offers unique advantages, with imaging-based

---

methods providing a high resolution for visualization and sequencing-based methods offering a significant depth for sequencing [9,10]. The data on spatial transcriptomes generated by these diverse technologies have a wide range of modalities, scales, and resolutions [11,12].

Deciphering the spatial domains and identifying the types of cells are prerequisites for investigating spatially resolved data on transcriptomics, and is commonly referred to as the task of spatial clustering [13,14]. The primary objective of this task was to classify the tissue sample into diverse sub-populations of cells, which in turn facilitates the analysis of the biological functions of clusters, tissue reconstruction, and cell–cell interactions. Approaches to spatial clustering can be categorized into two types based on their use of information on spatial location: spatial and non-spatial methods [15]. Such traditional techniques as Kmeans clustering and Louvain clustering rely solely on the profiles of gene expression to partition areas of the cells [16]. By contrast, The spatial location information could improve the clustering accuracy of the GNN-based method [17,18]. Giotto [19] leveraged both the count matrix and the physical coordinates to identify spatial domains by using the hidden Markov random field (HMRF) model. stLearn [20] applies a two-step procedure: It first uses the standard Louvain clustering workflow to handle the graph adjacency matrix and then leverages spatial positions to identify sub-clusters within broader clusters. SpaCell [21] uses a pretrained ResNet50 model and the ImageNet database to extract the features of images of each tile. It subsequently trains two autoencoders to obtain vectors in a latent space based on the features of the images and gene counts. BayesSpace [22] applies a fully Bayesian statistical model to enhance imaging resolution and clustering analysis. It assumes that nearby spots are in the same group in the given clustering structure based on a pre-defined spatial prior. SC-MEB [23] uses an efficient expectation maximization algorithm based on an iterative conditional mode to overcome the limitations of the smoothness parameter and the number of clusters in the Bayesian model. Despite the potential of the information on spatial location for improving the accuracy of clustering, the above-mentioned algorithms of spatial clustering have not achieved optimal performance [24,25].

Graph neural networks (GNNs) have become popular in the recent literature because they can leverage the profiles of gene expression and spatial information [26]. Researchers have proposed a variety of heterogeneous graph convolutions to use graph-related information, including the graph attention network (GAT), sample and aggregate (SAGE), simple graph convolution (SGC), graph convolutional network (GCN), topology-adaptive graph (TAG), and unified message-passing model (UniMP). Many studies have sought to apply these GNNs to generate the results of partitioning and identify spatial domains. For instance, SpaGCN [27] uses a GCN to identify spatial domains based on the aggregation of gene expressions, histology, and spatial location. The algorithm identifies spatially variable genes in each domain. The SEDR [28] exploits two networks to create a low-dimensional latent embedding of the data on spatial transcriptomics. One of the networks is a deep autoencoder for learning gene representations while the other is a variational graph autoencoder for learning spatial embeddings. The CCST [29] generates an embedding of cell nodes that contains information on both the spatial structure and the gene expressions. This method performs cell clustering for spatial transcriptomics by using a series of layers of the GCN that are embedded into a deep graph infomax (DGI) module. The authors of Ref. [30] used the conST algorithm to integrate gene expression, morphology, and spatial information by using a multi-modal contrastive learning framework. This helps maximize the information contained in the local and global graph contexts to generate effective latent embeddings. STAGATE [31] uses a framework of the graph attention-based autoencoder to identify the spatial domain. To determine the spatial positions of spots, a cell type-aware spatial neighbor network is used to characterize the spatial similarity between boundaries of the domain. The authors of Ref. [32] proposed DeepST, which uses a deep neural network to process morphological images and create a spatially augmented matrix by appending data on gene expression and spatial location. DeepST also applies two autoencoders to obtain a latent representation of the augmented data. GraphST [33] is a self-supervised graph-based technique of contrastive learning for spatial transcriptomics analysis that encompasses spatial clustering, multi-sample integration, and cell-type deconvolution. The contrastive learning framework in the spatial clustering module learns informative and discriminative representations of spots by minimizing the embedding distance between spatially adjacent spots. To the best of our knowledge, no comprehensive survey has analyzed and compared these advanced algorithms to date.

Several studies have provided overviews of emerging applications of Artificial Intelligence algorithms to spatial transcriptomics analysis. For example, the authors of Ref. [34] highlighted various problems in research on spatial transcriptomes, including data pre-processing, spatial clustering, spot deconvolution, gene imputation, the reconstruction of spatial location, and cell–cell interaction. They proposed the corresponding solutions while specifying the underlying assumptions. Ref. [35] provided a systematic overview of clustering-based approaches in computer science, big data, Artificial Intelligence (AI), and robotics, and categorized the corresponding algorithms into hierarchical and partitioning-based methods of clustering. The authors of Ref. [36] reviewed the deep clustering algorithm, which involves jointly optimizing representation learning and clustering. This review covered multiple definitions of clustering, including deep, shallow, hard, soft, partitioning-based, and overlapping clustering. Similarly, Ref. [37] discussed various applications of AI for data on spatial transcriptomes, including spatially variable gene detection, clustering analysis, communication analysis, spot deconvolution, and gene enhancement. Reference [38] focused on methods of calculation and challenges related to investigating and interpreting spatial data from a spatial perspective. This review addressed five important topics: exploration of spatial data, quality control and pre-processing, annotation of cell- and tissue-level data, tissue-wide spatial data interpretation, and prospective insights. Furthermore, the authors of Ref. [17] benchmarked and compared methods of cell-type clustering for spatially resolved transcriptomics data by focusing on several spatial clustering algorithms, including Seurat [39], Giotto [19], stLearn [20], SpaCell [21], and BayesSpace [22]. Reference [40] provided a synthetic review of clustering algorithms while outlining their strengths and weaknesses. This study covered traditional clustering algorithms based on partitioning, hierarchies, fuzzy theory, distribution, density, graph theory, grids, fractal theory, and models. The authors of Ref. [41] discussed machine learning methods that focus on data on spatial transcriptomes with histological images. They also considered research on clustering, spatially variable genes, deconvolution, enhancement, and cell–cell communication. The relevant algorithms included the HMRF model, SpaCell, SpaGCN, stLearn, and BayesSpace. Supplementary Table S1 summarizes the overall characteristics of the above reviews of clustering algorithms. However, these studies have not considered the most recent algorithms of spatial clustering for spatial transcriptomics, nor have they accounted for significant factors influencing clustering-related performance through correlation analysis. Improving the accuracy of clustering thus poses a challenge for future studies in the area.

To address the gaps in research in the area, the authors of this study provide a comprehensive overview of eight most recent approaches to identifying spatial domains in spatially resolved transcriptomics. These algorithms of spatial clustering, namely, conST [30], GraphST [33], STAGATE [31], DeepST [32], SpaGCN [27], CCST [29], SpaceFlow [42] and Spatial-MGCN [43] have been developed within the past 2 years, and are known for their efficiency and robustness that help them attain impressive clustering-related performance through mechanisms of unsupervised learning. Moreover, we extend these advanced frameworks of spatial clustering to encompass 60 clustering scenarios by considering such factors such as the candidate GNNs, techniques of downstream partitioning, PCA-based reduction, and refined methods of

improvement. We compare their accuracy of clustering, complexity of implementation, the required inputs, computational efficiency, robustness, runtime, and memory usage across multiple datasets of spatial transcriptomics to evaluate the above conditions of clustering. The datasets were obtained from different technologies of spatial transcriptomics, some of which had a ground truth while the others did not. This comprehensive comparison serves as a benchmark for the clustering algorithms. For the sake of clarity, Supplementary Table S2 presents the acronyms used in this article. We think that this overview can help practitioners and researchers in further exploring the use of heterogeneous methods of learning for clustering. We expect the performance of the algorithms of spatial clustering to undergo exponential growth with advancements in the field.

The remainder of this paper is structured as follows: We first provided a detailed introduction to the related datasets of spatial transcriptomics and methods of benchmarking, and then discussed the installation, use, defects, and merits of each technique as well as its extension to various clustering scenarios. Following this, we compared the most recent spatial algorithms from multiple perspectives, and applied them to various datasets of spatial transcriptomics, some with a golden standard and others without, to draw inter-related interpretations of their clustering-related performance. Finally, we provided directions for future research on spatial clustering, including the methods of learning and data fusion as well as potential motivations to drive further research. The structural organization of our survey is shown in Fig. 1.

2. Materials and methods

2.1. Algorithms of spatial clustering

The task of spatial clustering plays a crucial role in the analysis of data on spatial transcriptomics. During this process, cells/spots are grouped together based on the similarity between their gene expressions and profiles of spatial location. This step is essential for various tasks, such as inferring spatial trajectories, reconstructing tissue structures, understanding gene modules, and exploring cell–cell communication. However, enhancing the accuracy of clustering remains a challenging task. Owing to a lack of accurate and comprehensive surveys of recent algorithms of spatial clustering, research is needed to harness the

potential of learning-based approaches to improve clustering-related performance.

We focused on benchmarking the frameworks of spatial clustering released in the last two years: namely, GraphST, conST, DeepST, STA-GATE, CCST, SpaGCN, SpaceFlow and Spatial-MGCN. Remarkably, all of these frameworks are based on graph-based deep learning, and use a GNN. Because the choice of the GNN, algorithms of downstream clustering, PCA-based reduction, and techniques of refinement significantly impact GNN-based methods of clustering, we extend these eight spatial clustering frameworks to encompass 60 clustering scenarios. We provide a detailed introduction to these clustering frameworks and meticulously evaluate the generated clustering scenarios on multiple datasets of spatial transcriptomics to observe their clustering-related performance (Table 3).

SpaGCN [27] is a method that integrates the division of the spatial domain and the identification of spatially variable genes (SVGs) through a graph convolutional layer. It constructs a weighted, undirected subgraph that connects adjacent spots based on their spatial coordinates and the pixel values of their histological images. The profiles of gene expression are then fed into a GCN [44]. Clusters are generated iteratively in an unsupervised manner by using aggregated spot representations. Enriched SVGs or meta-genes are identified through the analysis of differential expressions (DEs) [45]. Finally, SpaGCN separates samples of spatial tissues and confirms the coherence of the related patterns of expression. Its performance has been verified on seven publicly available datasets, including the dataset of the olfactory bulbs of mice [46], the human dorsolateral prefrontal cortex (DLPFC) dataset [47] from 10x Visium, and the mouse hypothalamus dataset [48] from MERFISH. Its spatial clustering function has been compared against Louvain, stLearn, and BayesSpace clustering, while its SVG detection function has been compared with those of SpatialDE [49] and SPARK [50]. While SpaGCN outperforms these baselines, it is important to note that the latter are not currently considered to be state-of-the-art spatial algorithms.

Cell clustering for spatial transcriptomics (CCST) [29] is an unsupervised approach to cell clustering that uses the GCN to handle data on non-Euclidean spatial structures (June 2022). This technique converts the data on spatial structures into a graph, in which the nodes represent gene expressions and the edges indicate adjacency relationships. By leveraging a series of layers of the GCN embedded into a deep graph
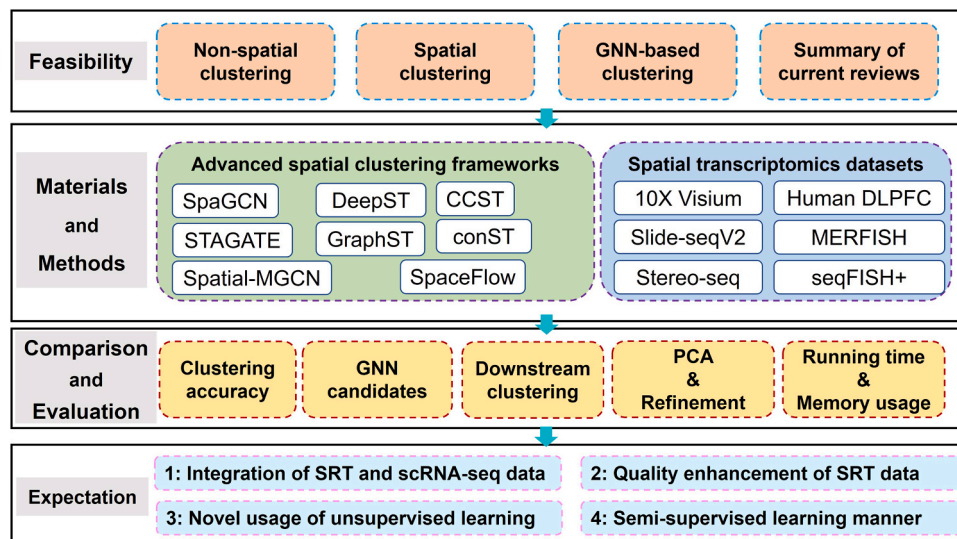

Fig. 1. The overall architecture of this review. We first analyzed the necessity and feasibility of this survey, and then we provided a detailed introduction to the most recent algorithms of spatial clustering developed in the past two years. Following this, we elucidated the datasets of spatial transcriptomics used in this study to assess the algorithms, and then comprehensively compare these benchmarking algorithms from multiple perspectives. Finally, we discussed the anticipated developments in spatial clustering for spatial transcriptomics along several directions.

infomax (DGI) module [51], CCST transfers positive embedding vectors of the cell nodes while obtaining the relevant negative embeddings from the graph. A discriminator based on the positive and negative embeddings is trained to encode the cell embeddings and detect the spatial regions. The performance of CCST has been evaluated on two datasets: FISH-based single-cell transcriptomics, and spot-based spatial transcriptomics (STs). The datasets used for its evaluation included MERFISH, seqFISH+ , DLPFC, and the human breast cancer dataset. CCST has also been compared with BayesSpace, SpaGCN, SEDR, stLearn, Giotto, Seurat, and STEEL [52]. The results have shown that it has the potential to improve the accuracy of tissue identification and enhance our understanding of the spatial organization of cells.

STAGATE [31] is a comprehensive toolbox designed for analyzing spatially resolved transcriptomics (SRT) data by using an adaptive graph attention autoencoder (April 2022). This versatile toolbox can be used for practical analysis, including spatial clustering, visualization, inference of the spatial trajectory, data denoising, and 3D domain extraction. A key contribution of STAGATE is the cell type-aware spatial neighbor network (SNN), which can accurately characterize spatial similarity along the boundaries. Moreover, STAGATE introduces an attention mechanism that can adaptively learn the edge weights of SNNs. Downstream clustering algorithms, such as mclust [53] and Louvain, manage the updated spot representations. The datasets used to verify the performance of STAGATE include the human DLPFC, mouse brain [54], Slide-seqV2, Stereo-seq, and STARmap datasets [55]. It has demonstrated superior performance to non-spatial methods (e.g., Louvain from Scanpy [56]) and five spatial approaches (Giotto, BayesSpace, stLearn, SpaGCN, and SEDR). However, a comparison with more advanced clustering techniques is warranted to comprehensively assess its capabilities.

DeepST [32] (October 2022) is another GNN-based framework that can integrate morphological image tiles, gene expressions, and data on the spatial location from ST by using a domain-based adversarial technique. This approach uses two autoencoders to generate latent embeddings. One is a denoising autoencoder that generates the non-linear representation of gene expressions, while the other is a variational graph autoencoder (VGAE) [57] responsible for deriving graph embeddings. The graph adjacency matrix is computed by using the K-nearest neighbor (KNN) [58] method along with spatial coordinates. The clustering performance of DeepST has been thoroughly considered by comparing it with several popular techniques, including Kmeans, Seurat, stLearn, SpaGCN, SEDR, and BayesSpace, on the human DLPFC dataset. Furthermore, its capacity for generalization has been verified through tests conducted on disparate ST platforms, including 10x Visium, Slide-seqV2, Stereo-seq, MERFISH, and 4i. The Leiden algorithm from Scanpy has been applied to DeepST as the method of downstream clustering, and its resolution has been searched by using a step size of 0.01 when the number of clusters is unknown.

The conST [30] is an interpretable contrastive learning framework designed for analyzing spatially resolved transcriptomics data (January 2022). The process begins by extracting informative features from the morphology of each spot through masked autoencoders (MAE) [59]. Gene expression is then filtered by PCA and spatial coordinates are encoded through KNN. A VGAE module is subsequently used to generate graph embeddings based on a two-layer GCN. The network is supervised through contrastive learning, where mutual information is maximized at the local-local, local-global, and local-contextual levels [30]. Finally, the latent representations are obtained in an end-to-end manner and are refined by using a deep clustering method to ensure their compactness. conST has been evaluated on the mouse hypothalamus (MERFISH), mouse visual cortex (seqFISH), human DLPFC (10x Visium), human breast cancer (10x genomics), and datasets of the olfactory bulbs of mice (Stereo-seq). conST has also been compared with benchmarking methods of clustering, including Seurat, Giotto, stLearn, SpaGCN, SEDR, and BayesSpace. It is important to note that these baselines do not represent state-of-the-art clustering algorithms.

GraphST [33] is framework for the spatial clustering of ST data that uses contrastive learning (CL) (January 2023). The tasks of analysis in GraphST include spatial clustering, cell-type deconvolution, and multi-sample integration. By integrating GNNs and self-supervised contrastive learning [60] to minimize the distance between embeddings, GraphST can generate spot embeddings that improve its clustering performance. The spatial clustering module of GraphST begins by using spatial information to construct a neighborhood graph. A corrupted graph is then created by randomly shuffling vectors of gene expression across spots. The application of self-supervised contrastive learning in GraphST allows the spot embeddings to capture the local context while preserving their variability and the information on their neighborhood. This leads to the generation of positive and negative spot embeddings from the original and the corrupted graphs, respectively. Consequently, spatially adjacent spots exhibit similar embeddings while non-adjacent spots exhibit dissimilar embeddings. The trained model of GraphST is stable and delivers balanced performance. Its spatial clustering was assessed on five datasets: the human DLPFC (10x Visium), mouse brain (10x Genomics), human breast cancer (10x Visium), olfactory bulbs of mice (Stereo-seq), and mouse embryos datasets (Stereo-seq). It was compared with the STAGATE, BayesSpace, conST, SpaGCN, Giotto, and Seurat benchmarking methods to assess its performance.

SpaceFlow [42] utilizes graph convolutional encoders for Spatial Transcriptomic (ST) data analysis, generating a spatially coherent low-dimensional embedding, domain segmentation, and pseudo-Spatiotemporal Map (pSM) that represent tissue characteristics (July 2022). This involves constructing a Spatial Expression Graph (SEG) and utilizing a graph convolutional encoder within a Deep Graph Infomax (DGI) framework to produce consistent low-dimensional embeddings capturing spatial expression patterns. In the Human DLPFC dataset, SpaceFlow was subjected to clustering comparisons with Seurat, Giotto, stLearn, MERINGUE [61], and BayesSpace. The conclusive findings indicate that SpaceFlow outperformed the other five methods.

Spatial-MGCN [43] is the latest spatial clustering algorithm based on a multi-view Graph Convolutional Network (September 2023). It initially utilizes a multi-view GCN encoder to extract gene expression information, spatial information, and their combinations. Next, it employs an attention mechanism to adaptively fuse these components. Following this, a ZINB decoder [62] is used to reconstruct the feature matrix, capturing the global information of the raw spatial expression profile. Finally, a spatial regularization constraint is incorporated into the representation learning process to preserve spatial neighbor information. Spatial-MGCN demonstrates superior clustering performance within the Human DLPFC dataset when compared to CCST, STAGATE, BayesSpace, SpaGCN, SEDR, stLearn, and Seurat.

This review provides a summary of the most recently reported frameworks of spatial clustering, which are listed in Table 1. The hyperparameter settings of these frameworks are depicted in Table 3 for a clear comparison. The framework of learning and the structure of the GNN have become increasingly complex over time, and the availability of morphological or histological images is surprisingly not a critical factor in tasks of spatial clustering. Instead, the key elements that significantly influence the accuracy of clustering are the candidate GNNs, techniques of downstream clustering, PCA-based reduction, and methods of refined correction (Table 2), as mentioned in the previous section. However, these clustering algorithms are often sensitive to the parameter settings such that they cannot generate consistently reproducible results. To address this issue, this review provides a comprehensive comparison of state-of-the-art (SOTA) frameworks of spatial clustering, GraphST, conST, DeepST, STAGATE, CCST, SpaGCN, SpaceFlow, and Spatial-MGCN with a focus on their clustering performance. Furthermore, these frameworks are used to generate 60 clustering scenarios to determine the optimal definitions of the significant elements in them. The details of these 60 scenarios, which include five candidate GNNs, three clustering algorithms, two choices for PCA, and

Table 1

Summary of the spatial clustering frameworks discussed in this review. It is evident that the learning method and GNN construction become increasingly complex from the top to the bottom frameworks. These state-of-the-art algorithms have not been previously compared in the clustering field. This article aims to demonstrate their respective advantages and disadvantages, as well as variations.

Method	Abstract	Implementation	Requirements	Merit	Demerit	Language	Code link
SpaGCN	A toolbox for differential expression analysis and spatial domain identification	GCN	Gene expression, spatial location, and histology image	Two analysis functions, images usability	Low clustering accuracy and reproducibility	Python	https://github.com/jianhuupenn/SpaGCN
CCST	Only for cell type clustering, a Hybrid adjacency matrix to underline inputs	DGI and GCN	Gene expression and spatial location	DGI module application, easy achievement	Less validated datasets and Lower stability	Python	https://github.com/xiaoyeye/CCST
STAGATE	An overall toolbox for ST data analysis	Autoencoder, graph attention network	Gene expression and spatial location	Easy implementation, manifold functions	Lack discussion of parameters setting	Python	https://github.com/QIFEIDKN/STAGATE_pyG
DeepST	A deep learning and GNN-based spatial clustering algorithm	VGAE and GCN	Gene expression, spatial location, and histology (optional)	Three neural networks to handle spatial data, expandable capability	Too complex, reproducibility needs further validation	Python	https://github.com/JiangBioLab/DeepST
conST	A contrastive learning framework for spatial clustering	Contrastive learning, VGAE, GCN	Gene expression, spatial information, and morphology (optional)	End to end manner, interpretability, first application of contrastive learning	Low clustering accuracy, single function	Python	https://github.com/ys-zong/conST
GraphST	A graph self-supervised contrastive learning framework	Self-supervised contrastive learning, VGAE, GCN	Gene expression, spatial information	Higher accuracy, easy achievement, negative samples	Lower stability, lack interpretability	Python	https://github.com/JinmiaoChenLab/GraphST
Spatial-MGCN	Multi-view GCN with attention mechanism	VGAE, GCN	Gene expression, spatial information morphology	Attention mechanism, Spatial regularization constraint	Lower stability, interpretability	Python	https://github.com/cs-wangbo/Spatial-MGCN
SpaceFlow	Spatially regularized deep graph networks	Deep graph infomax (DGI), GCN	Pseudo-spatiotemporal map, gene expression, spatial coordinates	Robust domain segmentation, downstream analysis	Lower clustering accuracy, lack adaptation	Python	https://github.com/hongleir/SpaceFlow

Table 2

Extensional clustering situations based on the six SOTA spatial clustering frameworks. The GNN selections would influence the quality of the latent embedding. The spatial domains are discerned by the downstream clustering algorithms using the generated latent embedding. The PCA reduction and refinement step would further improve the clustering accuracy and performance.

Spatial clustering framework	GNN candidate			Histological images	Downstream clustering	PCA reduction	Refinement
SpaGCN	GCN#	GATv2		Necessary	Kmeans	×	KNN#
	SGC	TAG			Leiden#		
	SAGE				mclust		Radius
CCST	GCN#	GATv2		Unnecessary	Kmeans	√	KNN
	SGC	TAG			Leiden#		
	SAGE				mclust		Radius
STAGATE	GCN	GATv2#		Unnecessary	Kmeans	×	KNN
	SGC	TAG			Leiden		
	SAGE				mclust#		Radius
DeepST	GCN#	GATv2		Optional	Kmeans	×	KNN#
	SGC	TAG			Leiden#		
	SAGE				mclust#		Radius
conST	GCN#	GATv2		Optional	Kmeans	×	KNN
	SGC	TAG			Leiden#		
	SAGE				mclust		Radius
GraphST	GCN#	GATv2		Unnecessary	Kmeans	√	KNN
	SGC	TAG			Leiden		
	SAGE				mclust#		Radius#
Spatial-MGCN	GCN#	GATv2		Unnecessary	Kmeans#	×	KNN
	SGC	TAG			Leiden		
	SAGE				mclust		Radius
SpaceFlow	GCN#	GATv2		Unnecessary	Kmeans	×	KNN
	SGC	TAG			Leiden#		
	SAGE				mclust		Radius

Note: # indicates the default setting in the original spatial clustering framework, √ implies that the principal component analysis (PCA) reduction would be conducted while × means not.

two methods of refinement, are provided in Table 2.

2.1.1. Exploration of datasets

To ascertain the effectiveness of a framework of spatial clustering, it is imperative to evaluate it across datasets of spatial transcriptomics (ST). In this review, we compare the aforementioned methods by using heterogeneous data on ST obtained from diverse technologies, including 10x Visium/Xenium, Slide-seqV2, Stereo-seq, MERFISH, and seqFISH. These datasets exhibit variations in the number of genes, cells/spots, scales, and resolutions. Consequently, a single captured spot may consist of one to 10 cells (referred to as a "spot" throughout this review). The datasets contained data on tissue slices captured from the human brain,

mouse brain, the olfactory bulb of mice, human patients with breast cancer, mouse liver, and mouse embryo. They were categorized into two types for the clustering task: those with a ground truth and those without one. The clustering labels in datasets with a ground truth were compared with a gold standard to assess the accuracy of evaluation of the methods considered, while the labels of segmentation were matched with a public atlas on humans or mice for the datasets without a ground truth. Detailed information on the estimated eight ST datasets used in this survey is provided in Supplementary Table S3.

The LIBD human DLPFC dataset is frequently used in this study. The raw data, comprising gene expressions, spatial locations, and histological images, are accessible from the data repository of 10x Visium, and annotated labels can be obtained from the spatialLIBD program [63]. This dataset consisted of data on 12 tissue slices, each labeled as samples 151507–151676. The number of spots in these samples ranged from 3460 to 4789, and the depth of sequencing was used to measured 33,538 genes. Each sample comprised five or seven spatial domains consisting of DLPFC layers and white matter. Notably, samples 151507–151510 and samples 151673–151676 contained seven clusters, while samples 151669–151672 contained five groups. These groups had explicit and well-defined boundaries, which rendered this dataset highly suitable for assessing the accuracy of the clustering algorithms. We evaluated the benchmarking methods on these 12 samples. Furthermore, owing to the chronological arrangement of these layers, this dataset was also used to infer spatial trajectory of cell development [47].

The second dataset was derived from a coronal section of tissues in the brain of mice. The original data files are accessible from the portal of the 10x genomics dataset. The data were pre-processed by using Scanpy, where this included quality control, embedding and clustering, visualization, and the identification of marker genes and spatially variable genes (SVGs) [64]. Domain annotation, also referred to as the ground truth, was performed based on the Allen brain atlas and the gene expression atlas of the mouse brain [65]. This dataset could be conveniently accessed through the Squidpy [66] package by using the *squidpy. datasets* function. It contained 15 annotated clusters, 2688 spots, and 18, 078 genes. The distribution of the spots was cluttered as each typically contained five to 10 cells. The five cortex layers were clearly demarcated in these data. The associated cropped data, comprising 704 spots and 16, 562 genes, were also stored in Squidpy to facilitate analysis. For the purposes of this review, the annotated version of this dataset in Squidpy served as the ground truth for evaluating the clustering labels generated by different algorithms.

The third and fourth datasets contained information on tissues in the olfactory bulbs of mice [67]. Data on ST for these tissues were generated by using the Slide-seqV2 and Stereo-seq techniques. Slide-seqV2 [7], an advanced version of the Slide-seq method, incorporates improvements in bead synthesis, array indexing, and library generation. Due to its ability to generate data on ST with near-cellular resolution and a high depth of sequencing, Slide-seqV2 was used to examine specimens of the olfactory bulbs of mice, specifically by using the sample Puck_200127_15. The dataset comprised 20,139 spots and 21,220 genes. Slide-seqV2 technology generated more spots than 10x Visium owing to its higher resolution.

Spatially enhanced resolution omics sequencing, Stereo-seq [8], is a cutting-edge technique that combines the captured RNA of tissues and nanoball patterned arrays of the DNA to achieve data on ST with a high resolution and a large field of view. Stereo-seq technology was applied to examine the olfactory bulbs of mice in Ref. [8], and yielded 19,527 spots and 27,106 genes. The number of genes detected in Stereo-seq surpasses that in Slide-seqV2. Both these datasets have been carefully arranged and discussed in the STAGATE framework, with the relevant links accessible in the package tutorial. These two datasets lack their respective ground truths. However, distinct areas of the data are demarcated by using specific genes and the Allen mouse brain atlas [68] to enable comparisons of algorithms of spatial clustering by comparing the regions and the marked genes.

The fifth dataset was derived from tissues of human patients of breast cancer, and is accessible through the repository of 10x Visium. This dataset is useful for analyzing the heterogeneous and immune microenvironments of tumors because the tissues exhibit high intratumoral and intertumoral differences. To assist in clustering, the sample was segmented into 20 regions based on the pathological features and gene expressions by using the SEDR [28] package. These annotated areas served as the baseline for the evaluation of clustering. This dataset consisted of 3798 spots and 36,601 genes. The authors of the original paper on the SEDR package compared it with Seurat, stLearn, and SpaGCN on this dataset. However, recently developed algorithms of spatial clustering like GraphST, STAGATE, and conST have not been evaluated on this dataset.

The innovative technology, 10x Xenium, presents a pioneering approach for the integration of single-cell, spatial, and in situ analyses of formalin-fixed paraffin-embedded (FFPE) tissue. The six breast cancer tumor datasets associated with this technology were systematically reprocessed and subsequently republished on December 6, 2022 [69]. Leveraging its non-destructive workflow, Xenium facilitates the spatial registration of RNA, protein, and histological data, consolidating them into a unified image. The robust capabilities of Xenium empower researchers to achieve spatial resolution at the single-cell level, allowing for the identification and characterization of 17 distinct cell types within the breast cancer tumor. This analysis encompasses a total of 164,079 cells and utilizes a 313-plex gene panel. To mitigate computational demands, a segmentation of this extensive dataset is implemented for the spatial clustering study. This refined dataset includes 15 cell types, incorporating 11,996 cells and maintaining the original 313-gene panel.

seqFISH represents an additional methodology offering single-cell resolution and employing image-based spatial transcriptomics [70]. The seven spatial transcriptomics dataset under consideration comprises tissue sections from mouse embryos at the 8–12 somite stage, targeting 387 specific genes. To elucidate cell fate decisions, seqFISH computationally integrates single-cell genomics data with high-resolution, spatially-resolved gene expression maps. This integration involves the recognition of cell types through the amalgamation of spatial context and data from two single-cell transcriptome atlases. Within the scope of this study, a pre-processed subset of seqFISH data, encompassing 19,416 cells and 351 genes, is obtained from Squidpy. The spatial map reveals the presence of 22 distinct cell types within the mouse embryo tissue.

MERFISH represents a highly multiplexed single-molecule imaging technology [71], capable of quantifying and documenting the copy number and spatial distribution of RNA species at the level of individual cells. Numerous MERFISH datasets are available within the Vizgen data program, including the MERFISH mouse liver map and the MERFISH mouse brain receptor map. This study specifically focuses on the pre-processed MERFISH mouse liver dataset obtained through Squidpy. The selected Liver1Slice1 dataset encompasses 364,235 cells, profiles 347 genes, and identifies 28 distinct clusters. Similarly, to enhance computational efficiency, a spatially focused subset of the original data is extracted for spatial domain analysis, comprising 13,806 cells, 347 genes, and 28 clusters.

2.1.2. Pre-processing the datasets

GNN-based algorithms of spatial clustering leverage profiles of gene expression and information on spatial location to execute tasks of spatial clustering, as illustrated in Fig. 2. The inputs to this framework are the feature matrix and the adjacency matrix (Fig. 2 A) that are generated during data pre-processing. Up to 3000 highly variable genes were selected by using Scanpy for the feature matrix. If the data on ST contained fewer than 3000 genes, all available genes were included. The adjacency matrix was obtained based on the distances between neighboring spots. Fig. 2 outlines several critical factors that can influence the performance of the algorithm for spatial clustering, such as the candidate GNNs (Fig. 2 B), algorithms of downstream clustering (Fig. 2 E), PCA-based reduction (Fig. 2 C), and methods of refinement (Fig. 2 F).
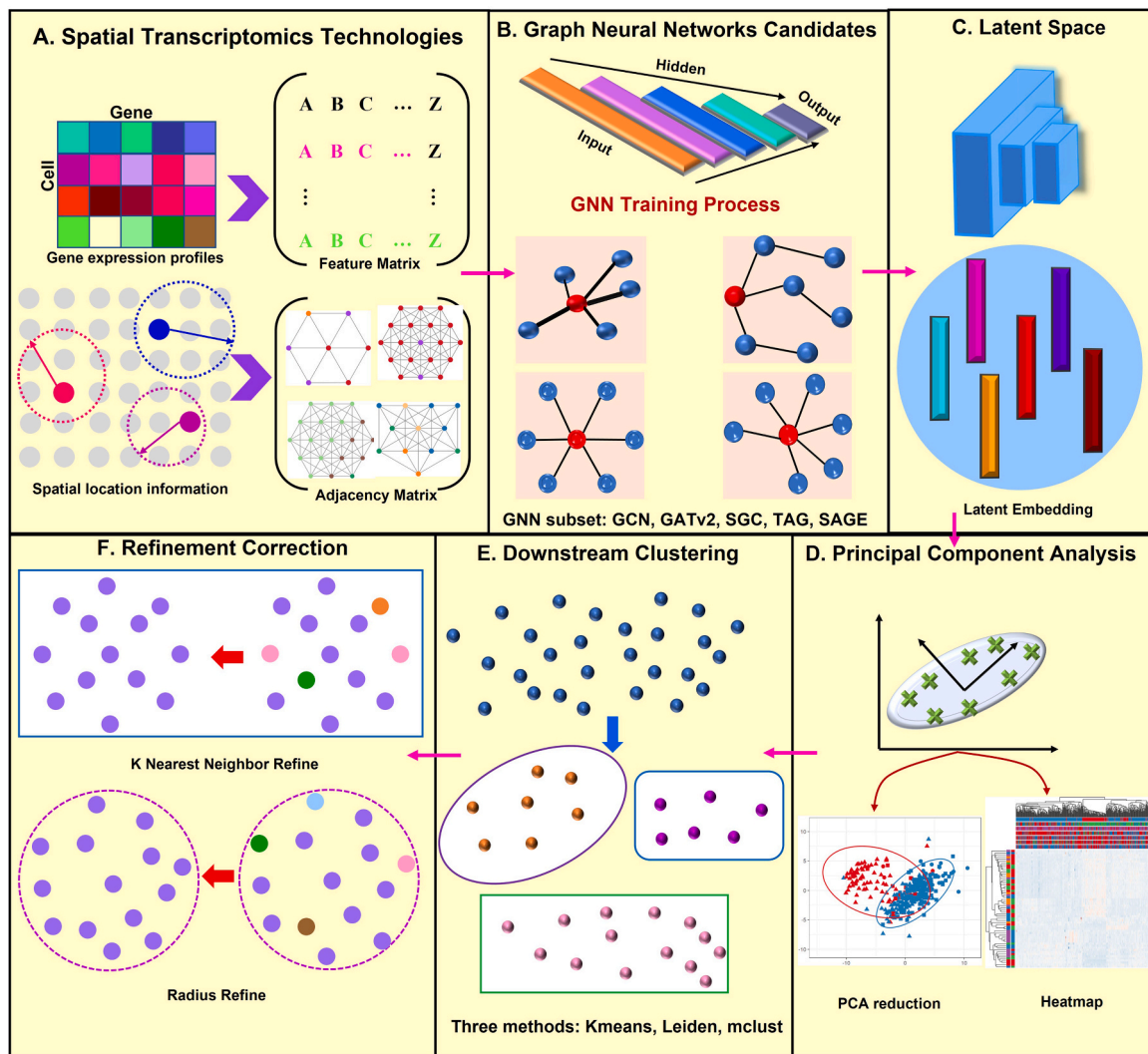
Fig. 2. Diagram of the workflow for the spatial clustering of spatially resolved transcriptomics data. Spatial transcriptomics (ST) data provide profiles of gene expression along with information on spatial location to be fed into a graph neural network (GNN). The choice of GNN significantly influences the quality of the generated latent embedding. For a thorough analysis of the data, principal component analysis (PCA)-based reduction may be applied to the latent embedding to reveal important elements. Algorithms of downstream clustering can be subsequently used to identify the spatial domains based on the latent embedding. The clustering labels were refined to reduce randomness. We examined five candidate GNNs, two PCA operations, three algorithms of downstream clustering, and two methods of refinement in this study for analyzing the ST data. This yielded 60 clustering scenarios that are discussed and explored here.

These factors are thoroughly analyzed and discussed based on the results of experiments.

A histological image serves as an optional input for tasks of spatial clustering. Table 2 shows that SpaGCN, DeepST, and conST are algorithms of spatial clustering that can use this morphological information. However, because such images may not be available in some datasets, they remain an optional input. To enhance the accuracy of clustering, certain algorithms (e.g., DeepST) adjust the construction of the adjacency graph to correspond to changes in the dataset of ST. In this review, we adhere to the original, default method of generating the adjacency matrix for each framework of clustering. We also explore five candidate GNNs, three algorithms of downstream clustering, two options for PCA, and two methods of refinement (Table 2). The eight benchmarking methods were thus tested on 60 clustering scenarios. We identified the best clustering scenario and determined the means to improve clustering-related performance. We also compare the clustering performance of the methods in each scenario to identify the optimal configuration.

Manual annotation plays a crucial role in pre-processing each dataset of ST. We examined five datasets of ST (Supplementary Table S3 and

Fig. 3). Most datasets contained ground-truth annotations, excluding the datasets of the olfactory bulbs of mice derived from Slide-seqV2 and Stereo-seq. Supplementary Table S3 provides a summary of the number of clusters in each dataset, which ranged from five to 20. The varying number of clusters allowed us to evaluate the ability of each algorithm to handle datasets of varying scales. The ground truth for the DLPFC dataset was obtained from the spatialLIBD project while the datasets of the brains of mice were annotated by using the Squidpy package. The human breast cancer dataset was marked in the SEDR package, and the STAGATE algorithm contained the annotated version of the Slide-seqV2-based dataset of the olfactory bulbs of mice. The dataset of the olfactory bulbs of mice based on Stereo-seq was obtained from the corresponding study [8].

Histological images are provided in the form of hematoxylin and eosin (H&E)-stained images in several datasets on ST, and are typically characterized by colors ranging from dark purple to pink hues [72,73]. Although different domains and spots within the images may exhibit distinguishable color patterns, this characteristic has not been identified as a significant factor for improving clustering performance. Histological images were thus available solely in the datasets on the ST of human
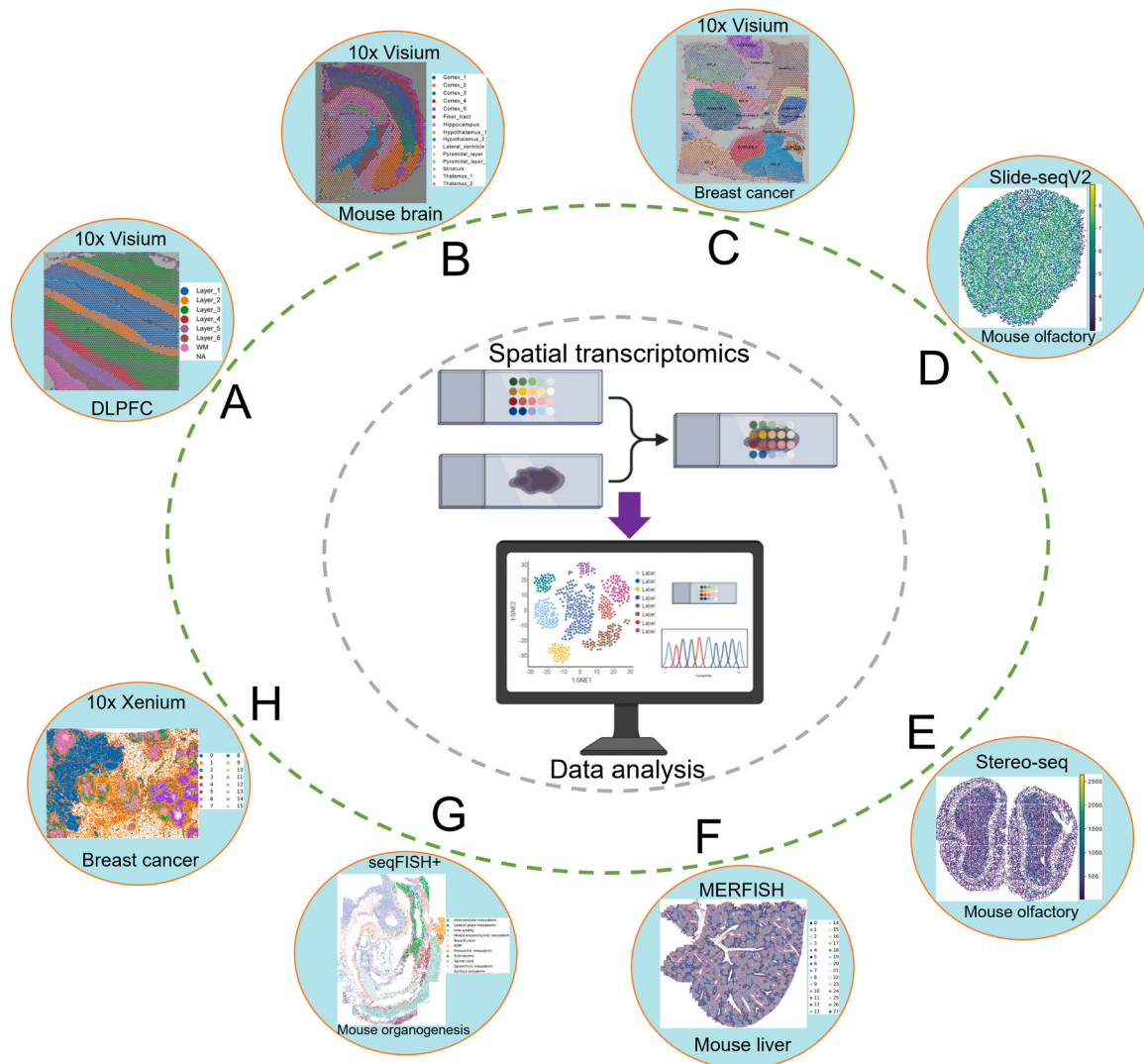
Fig. 3. This review considers the task of spatial clustering on eight spatially resolved transcriptomics datasets. **A.** The human DLPFC dataset consisted of 12 neuronal samples from three subjects, each with four sample sections. The displayed sample ID is 151510. **B.** The mouse brain dataset was obtained from 10X Visium, and each spot contained more than one cell. The entire dataset and its cropped section are available in the Squidpy package. **C.** The human breast cancer dataset was downloaded from 10X Visium, with manual annotations acquired from the SEDR package. **D.** The dataset of the olfactory bulbs of mouse was derived from Slide-seqV2 technology, and the particular section Puck_200127_15 was used. **E.** The Stereo-seq-based dataset of the olfactory bulbs of mouse was obtained from the original paper, and the special section was obtained from the results of SEDR analysis. **F.** The mouse liver dataset obtained from the Vizgen MERFISH Mouse Liver Map. The special slice is the Liver1Slice1 that measures 347 genes across over > 300,000 liver cells in a single mouse liver slice. **G.** The pre-processed subset of seqFISH data acquired from Squidpy package. **H.** The 10x Xenium dataset from the Xenium breast cancer tumor microenvironment Datbase. The related pre-processed data can be gained from Squidpy as well.

DLPFC, the brains of mice, and human breast cancer, all of which were retrieved from the data repository of 10x Visium.

2.2. Definitions of hyperparameters of each clustering algorithm

Each spatial clustering algorithm relies on graph-based deep learning and the GNN, which in turn requires tuning several hyperparameters to ensure optimal performance. We used the configurations to execute the eight benchmarking algorithms described in the corresponding original articles. We used the PyTorch Geometric (PyG) library [74] in Python to implement these frameworks of clustering. PyG provides a comprehensive range of methodologies for the development and training of GNNs on structured data, and is an ideal choice for our purposes.

Computing the adjacency matrix for SpaGCN involves defining the characteristic length scale, denoted by l. This was set to 0.5. Following the 2.2 computation of the matrix of gene expression, 50 principal components were used as the input to the GNN. The Louvain algorithm

[75] was then applied to the aggregated output matrix generate by the graph convolutional layer. In instances where the number of groups was not known, the resolution of the Louvain algorithm was varied between 0.2 and 0.1. The parameters of the network and the centroids of the clusters were optimized by using the Kullback-Leibler (KL) divergence loss [76]. To further enhance performance, SpaGCN used a refinement step based on the KNN to improve the clustering labels. A learning rate of 0.05, weight decay of 5e-3, random seed of 100, and 200 epochs were used during training.

The hyperparameter λ is crucial for balancing the gene expressions and spatial information of individual spots in CCST. The value of λ is changed with the resolution of spatial transcriptomics dataset. This is set to 0.3 for single-cell resolution data and, conversely, 0.8 for non-single-cell resolution data. Following the creation of the feature matrix, dimension reduction was performed by selecting 200 principal components. The DGI module was then applied to obtain an embedding vector, and the cell groups were identified by using Kmeans+ +. A total of 5000

epochs were used for training and the number of hidden channels was set to 256.

The generation of the adjacency matrix *A* in STAGATE depends on the dataset used. It can be generated by using either the KNN or the radius model. The KNN model was used for the data for 10x Visium, wherein the adjacency matrix comprised the six nearest neighbors. On the contrary, a radius was empirically determined for the other datasets to ensure that each spot had an average of six neighbors. While the cell type-aware SNN in STAGATE is an optional feature, the pre-clustering task was performed by using the Louvain algorithm with a resolution of 0.2. The default weight for the cell type-aware SNN is represented by the hyperparameter *α,* and was set to 0.5. The number of dimensions of the encoder were [30,512] during training, and it had a two-layer structure. The exponential linear unit (ELU) was used as the activation function [77]. The learning rate and weight decay were both set to 1e-4. STAGATE executes 500 iterations by default.

The DeepST method used 50 principal components to extract latent characteristics from the morphological images. The KD tree algorithm [78] was used to set the distance between spots by considering the 12 neighbors nearest to a given spot. To reduce the number of dimensions of the augmented data on gene expression, 100 principal components were used. A total of 1000 epochs were executed during training, with the number of dimensions of both the encoder and the hidden layers set to 64–16. The GCN was used as the GNN in DeepST.

The adjacency matrix *A* in the conST algorithm was computed by using the Euclidean distance, and the KNN was used to construct a graph based on the 10 neighbors closest to the given spot. The matrix of gene expressions was reduced to 300 dimensions through PCA. Morphological features were extracted by using a pre-trained MAE model with 768 dimensions and 100–20 hidden dimensions. The GCN had 32–8 hidden dimensions and a learning rate of 0.01 during the training of conST. The weights for the reconstruction and contrastive losses were 10 and 0.1, respectively, for the GNN. The weight decay was set to 0.01 and the number of epochs was 200.

The proximity between spots in GraphST was defined by using the Euclidean distance, and a graph was constructed by considering three neighbors nearest to each spot. The GCN in GraphST used the rectified linear unit (ReLU) [79] as the nonlinear activation function. The weights for reconstruction-related and contrastive loss were set to 10 and one, respectively. The encoder had 3000 input dimensions and 64 output dimensions during training, with a learning rate of 0.001. The chosen random seed was 41 and the weight decay was set to zero. The training of GraphST was conducted over 600 epochs. After having determined the spatial domains, a refinement step was applied to reset some labels by using a radius of 50 to define the number of nearest neighbors.

SpaceFlow preprocesses the gene expression data of the first 3000 Highly Variable Genes (HVGs) into a Spatial Expression Graph (SEG) using either alpha-complex-based or k-nearest-neighbor-based transformations. The Deep Graph Infomax (DGI) technique is then employed to encode the Spatial Transcriptomic (ST) data into low-dimensional embeddings representing cells or spots. Subsequently, the Leiden clustering algorithm is applied for clustering, with the parameter set to 50. During training, the model is parameterized as follows: it utilizes Parametric Rectified Linear Units (PReLU) as the activation function, employs the Adam optimizer with a learning rate of 0.001, and the maximum number of training epochs is set to 1000.

Spatial-MGCN computes gene expression similarity utilizing cosine distance metrics and assesses the proximity of neighboring spots based on their spatial coordinates within the tissue. Subsequently, these are integrated using a Multi-view GCN encoder. During the training phase, weight decay was set to 0, the learning rate was 0.001, and the number of epochs was 200.

The spatial accuracy of clustering can be influenced by various factors depending on the hyperparameters used in each algorithm. To evaluate the scalability of these benchmarking algorithms, we replaced the default GNN with four networks to generate varying latent

representations (Tables 2 and 3). We compared three methods of downstream clustering: Leiden, mclust, and Kmeans. PCA-based reduction was applied to the embedding vectors in CCST and GraphST. Moreover, the methods of refinement used in SpaGCN and GraphST were used to reset some clustering labels, thereby improving the accuracy of clustering. We thus analyzed the effects of both PCA-based reduction and the refinement step on the clustering-related performance of the algorithms.

2.3. Criterion of clustering for evaluation

The criterion of clustering was applied to quantify the clustering performance of each algorithm. This norm changed with the ground truth. When the ground truth was known, the adjusted Rand index (ARI) [80] was treated as the criterion of estimation, and was obtained by using the scikit-learn toolkit [81] by importing the ground truth and the predicted labels. These two vectors were denoted by $G = \{G_1, G_2, \ldots G_M\}$ and $P = \{P_1, P_2, \ldots P_M\}$, respectively. The ARI can then be described as

$$ARI = \frac{\sum_{i,j}\binom{n_{ij}}{2} - \left[\sum_i \binom{n_{i.}}{2}\sum_j\binom{n_j}{2}\right] \Big/ \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{n_{i.}}{2} + \sum_j \binom{n_j}{2}\right] - \left[\sum_i \binom{n_{i.}}{2}\sum_j\binom{n_j}{2}\right]\Big/\binom{n}{2}} \quad (1)$$

where *M* is the number of annotated clusters, $n_{i.}$ and n_j denote the numbers of spots belonging to P_i and G_j, and n_{ij} represents the number of spots located in P_i and G_j. The ARI ranges from zero to one, and a higher value reflects a higher accuracy of clustering.

In addition to ARI, the silhouette coefficient (SC) score, Davies-Bouldin (DB) index [82], the Normalized Mutual Information (NMI) and the Cell Stability Score (CSS) [83] were used to evaluate different aspects of clustering performance. The SC score of the predicted labels was calculated according to the mean intracluster and intercluster distances. It ranges from -1–1, and indicates the degree of dispersion between clusters. A higher SC score reflects a higher accuracy of clustering. The DB score is the average measure of similarity between each cluster and the cluster most similar to it. It is set in the range $[0, +\infty]$, and a lower DB index is preferable. The NMI metric assesses clustering algorithm performance by measuring the similarity between clustering results of two datasets, with higher values indicating increased consistency. The CSS is founded upon the Jaccard Index, a metric that measures the similarity between two sets. These four values were also determined by using the scikit-learn package.

3. Comparative results

3.1. Benchmarking eight spatial clustering frameworks on human DLPFC dataset

We evaluated the performance of eight frameworks of spatial clustering, namely, GraphST, conST, DeepST, STAGATE, CCST, SpaGCN, SpaceFlow and Spatial-MGCN, on the human DLPFC dataset. This dataset comprised 12 sections from three subjects, and each had been manually annotated. To ensure the reliability of the results, we ran each algorithm 10 times on each of the 12 samples. The clustering criterion used was the ARI and NMI. We also used three samples from each subject (namely, 151510, 151672, and 151676) to evaluate the ability of each algorithm to identify the spatial domain.

Fig. 4 shows the mean and variance of ARI and NMI for each benchmarking method. GraphST recorded the highest average ARI and the lowest variance, indicating good performance and stability. Spatial-MGCN also yielded promising results, closely following GraphST. Of the other six approaches, CCST had the largest mean ARI and the greatest variance, with its ARI values fluctuating significantly across diverse samples. DeepST, which used three neural networks to derive the latent

Table 3
Summary of the training properties for eight spatial clustering methods in this review. This table provides the default graph constructions and hyperparameters in the training process of eight compared approaches.

Methods	GNN type	Loss function	Number of layers	Action function	Optimizer	Weight decay	Learning rate	Epochs	PCA	Clustering method
SpaGCN	GCN	Kullback-Leibler divergence	1	ReLU	Adam	5e-3	0.005	200	50	Leiden
CCST	GCN	Negative sampling	4	RPeLU, sigmoid	Adam	0	1e-6	5000	30	Leiden
STAGATE	GATv2	Mean squared error	4	sigmoid	Adam	1e-4	0.001	1000	×	mclust
DeepST	GCN	Mean squared error, Binary cross entropy error, Kullback-Leibler divergence	3	ReLU	Adam	1e-4	5e-4	1000	×	Leiden
conST	GCN	Binary cross entropy error, Kullback-Leibler divergence	3	sigmoid	Adam	0.01	0.01	200	×	Leiden
GraphST	GCN	Binary cross entropy error, Mean squared error	2	ReLU	Adam	0	0.001	600	20	mclust
Spatial-MGCN	GCN	Consistency loss, Regularization loss	3	sigmoid, Softplus	Adam	0	0.001	200	×	Kmeans
SpaceFlow	GCN	Kullback-Leibler divergence	2	RPeLU	Adam	0	0.001	1000	×	Leiden

Note: PCA: Principal Component Analysis; "×" indicates that the method did not perform PCA dimension reduction.
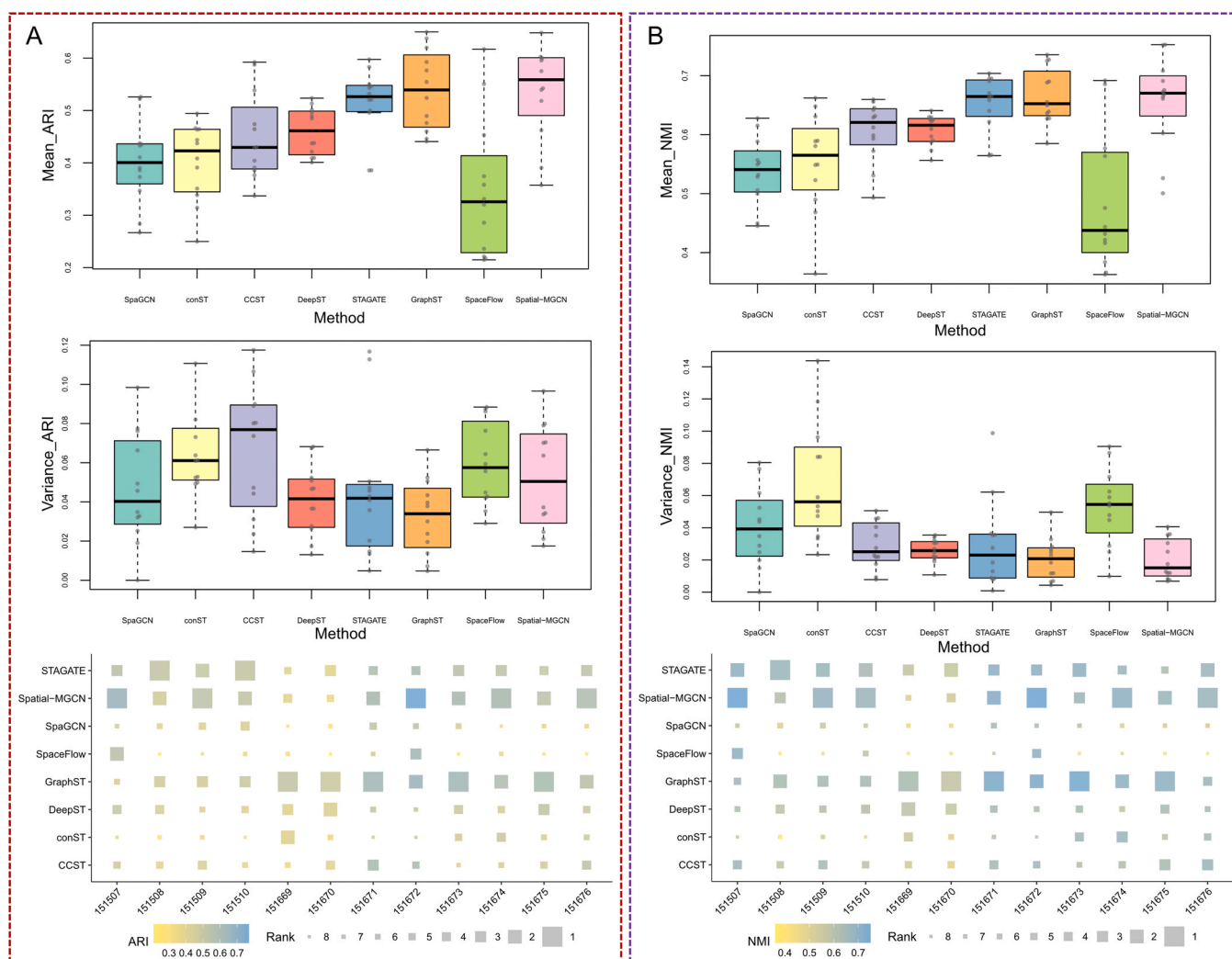
Fig. 4. Clustering performance of eight benchmarking methods on the human DLPFC dataset. **A.** The mean adjusted Rand index (ARI) of each clustering method was determined (Upper). The variance of each clustering method. GraphST had the highest average ARI and the lowest variance (Middle). The ARI of each spatial clustering algorithm was calculated on 12 samples of the human DLPFC. The eight methods are ranked from 1 to 8 in order from the highest to the lowest ARI (Lower). **B.** The Normalized Mutual Information (NMI) of each clustering method was determined (Upper). The variance of each clustering method. Spatial-MGCN had the highest average NMI and the lowest variance (Middle). The NMI of each spatial clustering algorithm was calculated on 12 samples of the human DLPFC. The eight methods are ranked from 1 to 8 in order from the highest to the lowest NMI (Lower).

embeddings, ranked in the middle in terms of performance. conST and GraphST, both of which were constructed through contrastive learning, demonstrated satisfactory performance, with GraphST outperforming conST. By contrast, SpaGCN, which required histological images, exhibited the lowest accuracy of clustering. Fig. 4 (Lower) shows the ARI score and NMI of each method on each sample. Rank 1 indicates that the corresponding approach had the highest ARI or NMI score. This figure confirms that SpaceFlow and SpaGCN were less accurate than the other six algorithms on the DLPFC dataset.

We also selected three samples from each subject to illustrate the performance of the above methods in terms of identifying the spatial domain. Fig. 5 presents the visualization of the ARI and NMI values of all methods on these selected samples. GraphST, STAGATE, and Spaital-MGCN outperformed the other five methods, where this is consistent with the average ARI and NMI values shown in Fig. 4 A. Conversely, SpaGCN and conST displayed the weakest performance on these samples. The clusters identified on the spatial coordinates are presented in Fig. 6. Samples 151510 and 151676 contained seven groups, while

sample 151672 had only five clusters according to the ground truth. The results of identification of the spatial domain generated by GraphST, STAGATE, and Spatial-MGCN closely aligned with the manual annotations, as indicated by their high ARI and NMI scores in Fig. 5. On the contrary, the identified group borders in SpaGCN and conST were unclear, because of which they had the lowest ARI scores on these samples. DeepST and CCST yielded middle-of-the-pack performance as reflected by their ARI and NMI values and outcomes of spatial clustering.

We used the identified spatial domains to generate plots of the uniform manifold approximation and projection (UMAP) [84] and infer the trajectories of spatial developmental. The partition-based graph abstraction function in Scanpy [56] was used to create the spatial trajectories, as depicted in Supplementary Fig. S1. The embedding of SpaGCN was unavailable, because of which we excluded it from consideration. Of the remaining methods, CCST and Spatial-MGCN exhibited the best spatial trajectories. Supplementary Fig. S2 shows the ARI and NMI values of the clustering methods on the remaining nine sections, while Supplementary Fig. S3 illustrates the interrelated
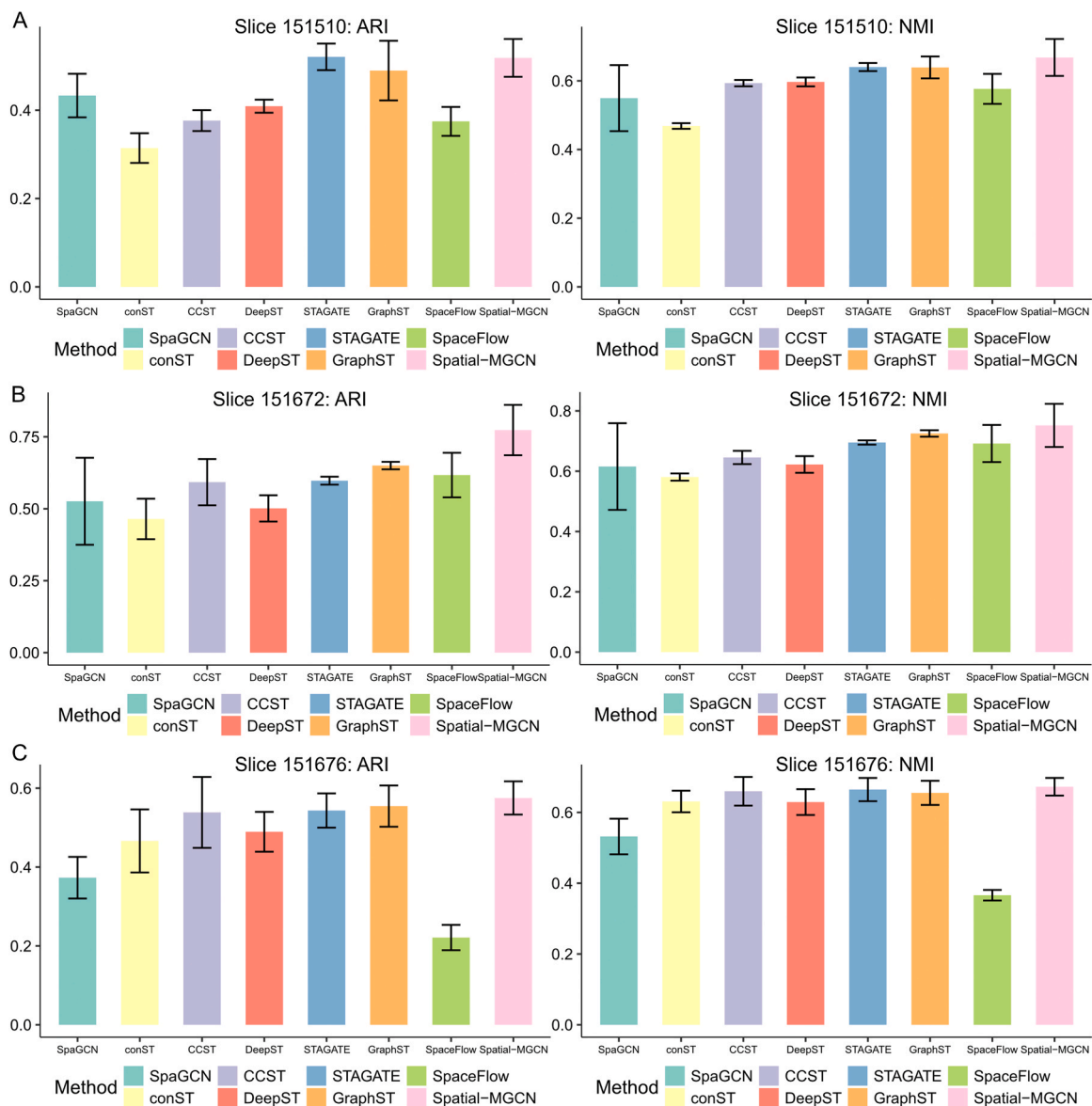

Fig. 5. The ARI and NMI scores of eight spatial clustering methods on three selected DLPFC samples. **A, B, C.** Three samples were obtained from three different subjects participating in the human DLPFC project. Each subject provided four sections of human brain tissues, and we selected one section from each to compare their ARI values and NMI. Samples 151510 and 151676 were annotated manually into seven clusters, while sample 151672 contained only five groups in the ground truth. The black segment visually represents the standard deviation.
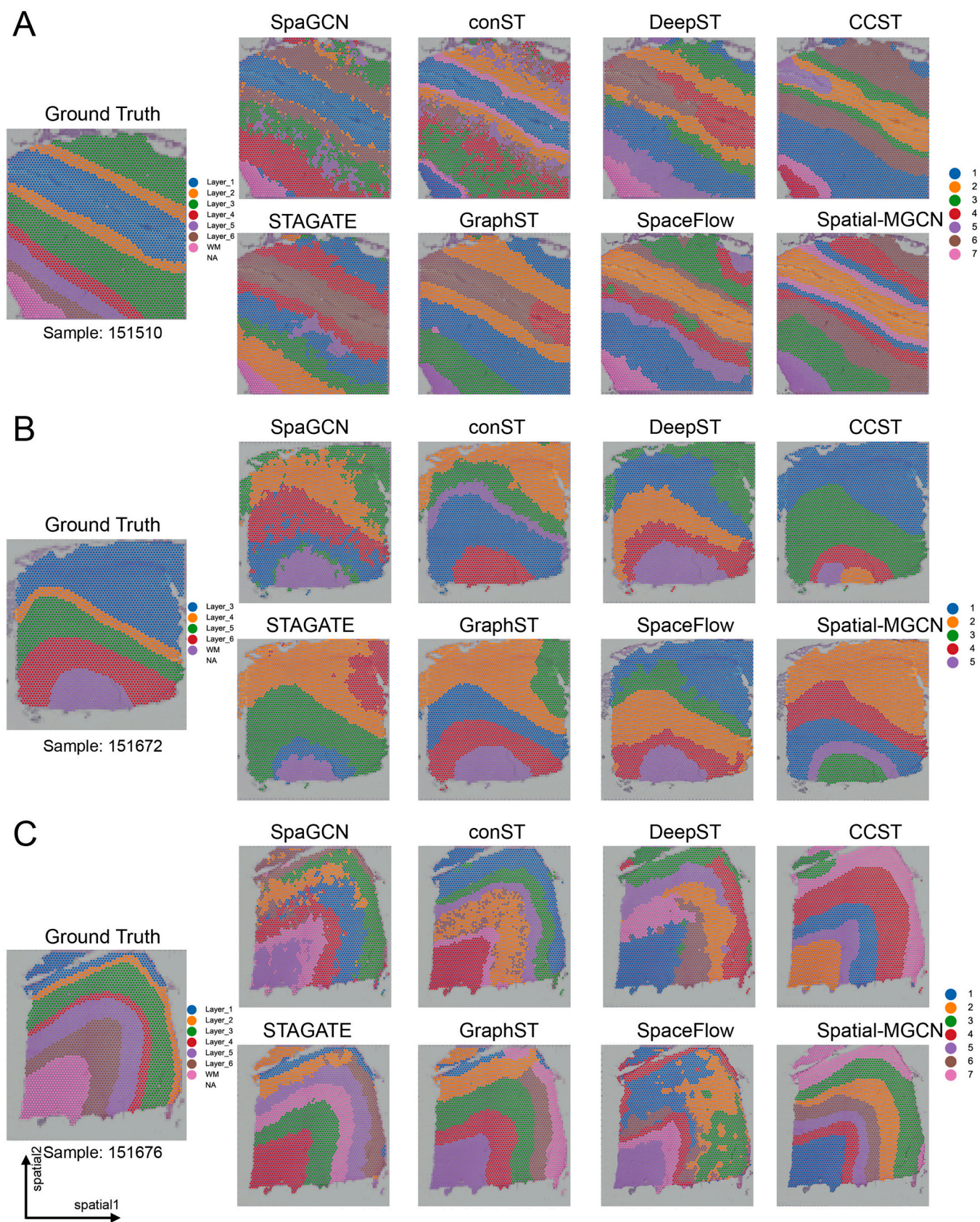
Fig. 6. The identified spatial domains of eight compared methods on three selected slices. **A, B, C.** The cell-type clustering task was performed on three samples (151570, 151672, and 151676) by using eight spatial clustering methods: SpaGCN, conST, DeepST, CCST, STAGATE, GraphST, SpaceFlow and Spatial-MGCN. These samples corresponded to the three subjects in the human DLPFC dataset. A comparison was made with the ground truth to assess the clustering performance of each baseline.

identified spatial regions. These ARI scores, NMI values, and identified spatial regions shed light on the overall clustering performance of the eight benchmarking approaches on the human DLPFC dataset. We classified the methods into three levels: STAGATE, GraphST, and Spatial-MGCN were in the first echelon, DeepST and CCST were in the second echelon, and SpaGCN, conST, and SpaceFlow were in the third echelon.

Furthermore, we applied SC score and DB index to evaluate the spatial connectivity of the segmented domains for each technique. The CSS is employed to explore how the stability changes if a perturbation is induced in the spatial transcriptomics dataset (Supplementary Fig. S4). The SC scores and DB indices are calculated on the 12 slices of human DLPFC dataset. A higher SC score and lower DB index indicate a better clustering accuracy. As a result, conST and CCST have a better performance of spatial connectivity (Supplementary Fig. S4A and S4B). For the special sample 151676, a 20% random deletion of spots is performed and the relevant CSS is computed. This metric implies that GraphST, CCST, and Spatial-MGCN have a better stability among these benchmarking methods.

3.2. Choice of GNN influences accuracy of spatial clustering algorithms

In this section, we investigated the impact of the GNN used (Fig. 2 B) on the performance of the eight methods in terms of spatial clustering. The choice of GNN influences the quality of the generated latent representations. The experimental data were derived from the dataset on tissues of human patients of breast cancer, and were measured using 10x Visium and annotated by the SEDR package. A total of 20 clusters were considered. The original GNN used in SpaGCN, CCST, DeepST, conST, and GraphST was the same, i.e., GCN. However, STAGATE uses GATv2 [85] as its primary GNN. To construct a set of candidate GNNs, we empirically added three other GNNs (SGC [86], TAG [87], and SAGE [88]), as shown in Table 2. The default algorithms of downstream clustering identified the spatial domains based on the derived latent embeddings by using the five GNNs. Their ARI values were then calculated based on the predicted labels and manual annotation.

The results in Fig. 7 show the ARI values of the five GNNs used in each method. The GNNs were ranked in ascending order based on their ARI values, with the original GNN highlighted in red. Surprisingly, the original GNN did not yield the highest ARI in any of the five methods except for GraphST. DeepST achieved the highest ARI of 0.618 with the SGC, slightly outperforming its original GCN (0.613) (Fig. 7 F). SpaGCN
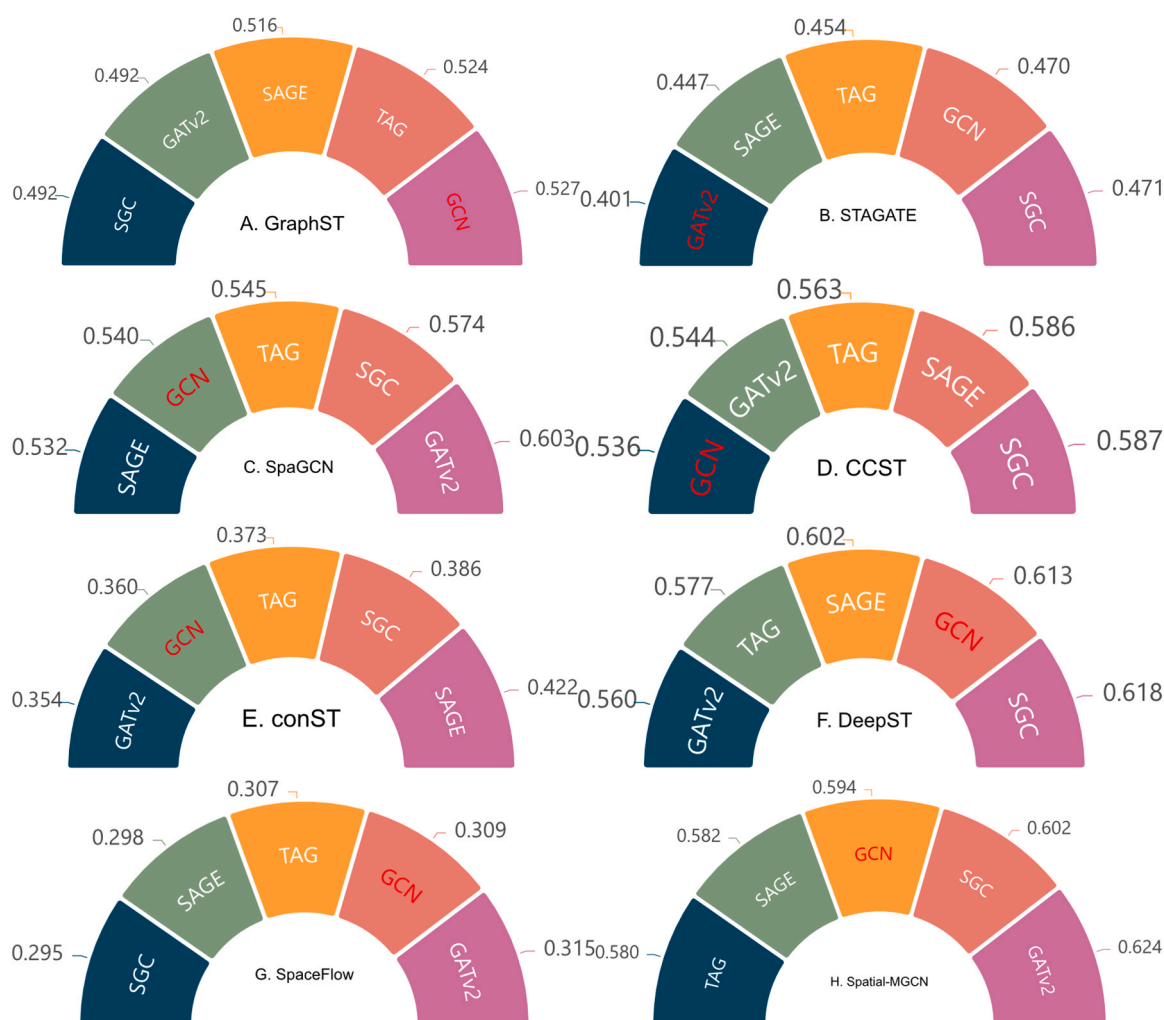

Fig. 7. Impact of the choice of graph neural network on various frameworks of spatial clustering. **A.** The ARI values of five GNNs in GraphST when applied to the human breast cancer dataset. **B.** The influence of the five GNNs on clustering-related performance in STAGATE. **C.** ARI scores of GATv2, TAG, SAGE, GCN, and SGC when using the SpaGCN framework. **D.** ARI values in CCST. The GNN given in red denotes the original GNN for each spatial clustering algorithm. **E.** ARI scores with respect to the five GNNs in the conST method. **F.** ARI values of five GNNs in DeepST. The primitive GNN was the GCN in this method. SGC was the best choice for DeepST on the human breast cancer dataset. **G.** ARI scores with repect to five GNNs in SpaceFlow framework. The GNNs are ordered based on their ARI values. **H.** Five GNNs have different ARI values in Spatial-MGCN framework.

obtained its best ARI score of 0.603 with GATv2, representing an improvement of 11.7% over its original GCN (0.54). The highest ARI scores of conST (0.422) and STAGATE (0.471) were both lower than 0.5. Moreover, the optimal GNNs in these two methods were not their original ones. Furthermore, replacing the GCN (0.536) with the SGC (0.587) in CCST led to a 9.5% improvement in its performance (Fig. 7 D). The best GNN for CCST, DeepST, and STAGATE was the SGC. These results suggest that the choice of GNN can significantly impact the accuracy of clustering, and the original GNN may not always be the best option.

Fig. 8 illustrates the spatial domains identified by the eight benchmarking spatial clustering algorithms. The latent embeddings were generated by using the best GNN for each method (Fig. 7). For instance, the SGC and GATv2 were used as the GNNs in DeepST and SpaGCN, respectively, and led to ARI scores higher than 0.6 for both techniques. Notably, some clusters in these two methods exhibited similarities. For example, group 0 in DeepST corresponded to cluster 1 in SpaGCN, while cluster 2 in SpaGCN aligned with group 1 in DeepST. On the contrary, the distribution of spots in conST and STAGATE appeared to be more intricate, as indicated by their lower ARI values. The SGC was the best GNN in CCST, DeepST, and STAGATE. However, despite using the same GNN, these methods exhibited diverse spatial regions due to the influence of the algorithms of downstream clustering, PCA-based reduction, and methods of refinement. Thus, the choice of GNN significantly impacted their clustering performance, and the default GNN used in these benchmarking methods may not be optimal.

3.3. Effects of downstream clustering methods on clustering performance

Following the application of the GNN to generate the latent embeddings, we used methods of downstream clustering to spatially partition the areas of the tissues (Fig. 2 E). In this section, we evaluated the impact of three methods of clustering, namely, Kmeans, Leiden, and

mclust, on the performance of the methods in terms of spatial clustering. These techniques are also summarized in Table 2. Kmeans and Leiden were implemented by using the scikit-learn toolkit [81], while mclust was implemented using R language with the associated software [89]. The dataset used for estimation was a coronal section of the mouse brain, and its pre-processed version with annotated clusters was accessed from the Squidpy package [66]. The process of annotation relied on the Allen brain atlas and the atlas for the gene expression of the brains of mice, and yielded a total of 15 clusters. The ARI score was computed by comparing the predicted labels with the manual annotation. It is important to note that SpaGCN was not considered in this analysis as the latent embedding for it was unavailable.

Fig. 9 displays the ARI values of the three approaches to downstream clustering when used in each method. The methods marked in red correspond to the original choice in each spatial clustering framework. For instance, the primary method of clustering used in DeepST, GraphST, and STAGATE was mclust, while CCST, SpaceFlow, and conST used Leiden by default. Spatial-MGCN employed Kmeans to implement the downstream clustering domains. The highest ARI value was recorded by DeepST with mclust (ARI = 0.63, Fig. 9 A) and GraphST with Leiden (ARI = 0.63, Fig. 9 D). The optimal methods of clustering for DeepST, CCST, conST, GraphST, STAGATE, SpaceFlow, and Spatial-MGCN were mclust, Leiden, mclust, Leiden, Kmeans, mclust, and mclust, respectively. Notably, the best method of downstream clustering varied across these clustering frameworks. The accuracy of clustering of conST, GraphST, STAGATE, SpaceFlow, and Spatial-MGCN could be improved by replacing their clustering algorithms. In particular, replacing mclust in STAGATE with Kmeans led to an 11.5% improvement in its accuracy of clustering (Fig. 9 E).

Fig. 10 illustrates the regions of the tissues on the spatial coordinates for each clustering framework. The ground truth profiled by Squidpy is presented in Fig. 10 A. The results of spatial clustering were obtained by using the best clustering method. It is apparent that the spots in conST,
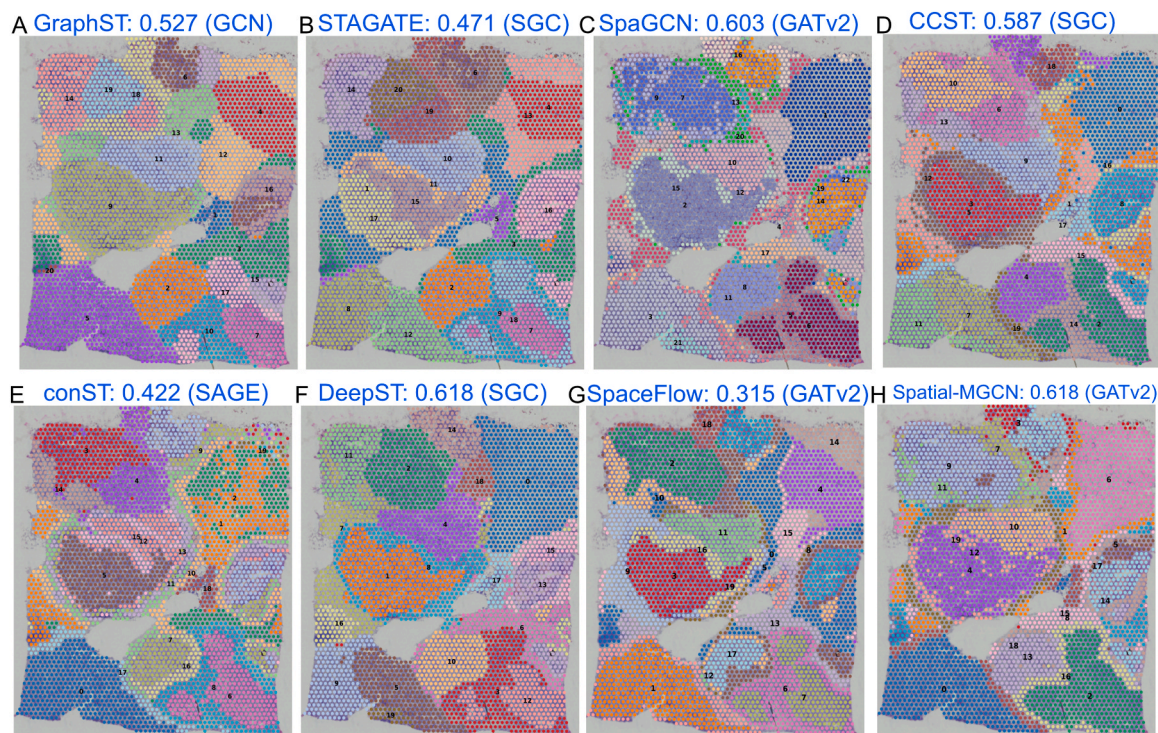

Fig. 8. The results of cell-type identification when using eight methods with the best-performing GNN. **A, B, C, D, E, F, G, H.** The data on spatial transcriptomics from the human breast cancer dataset as measured by 10x Visium technology. This data were annotated in the SEDR package with 20 regions. DeepST and Spatial-MGCN recorded the highest ARI (0.618) with the SGC and GATv2, respectively, while their original GNN (GCN) yielded an ARI of 0.613 and 0.594. When a new GNN was used, seven methods (CCST, conST, DeepST, SpaGCN, STAGATE, SpaceFlow, and Spatial-MGCN) had higher ARI scores than the corresponding original methods with the default GNN.
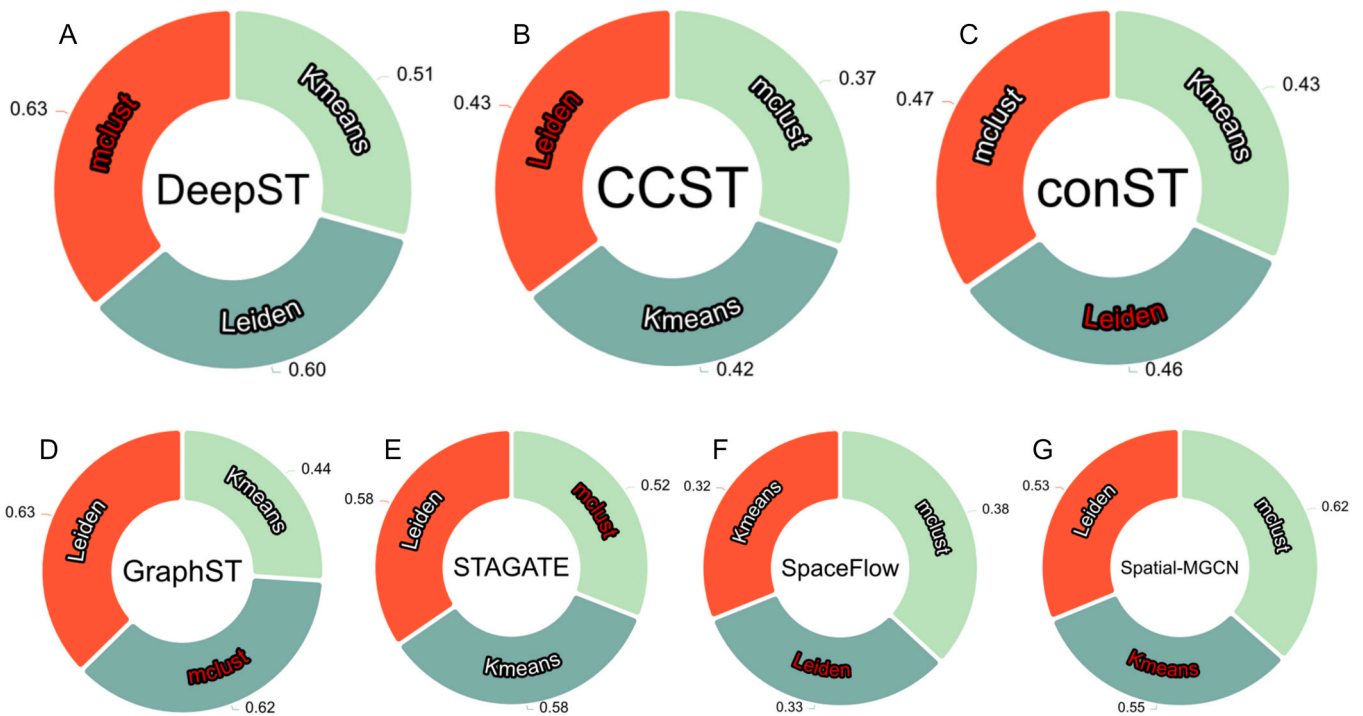
Fig. 9. The impact of three downstream clustering methods on five spatial clustering algorithms. **A.** The ARI values of DeepST with respect to mclust, Kmeans, and Leiden. The data were from a mouse brain dataset provided by the Squidpy package. **B.** The ARI scores of Leiden, Kmeans, and mclust in CCST. The method given in red indicates the original technique in each framework. **C.** Leiden was the primary method of clustering in conST. **D, E.** The original clustering algorithm in GraphST and STAGATE was mclust. However, the accuracy of clustering of Leiden was higher in these two methods. **F, G.** The primary downstream clustering method in SpaceFlow and Spatial-MGCN is Leiden and Kmeans, respectively. Different clustering approaches have various clustering accuracy. DeepST and GraphST achieved the best clustering performance, with an ARI of 0.63 on the mouse brain dataset. The best clustering methods in conST, GraphST, STAGATE, SpaceFlow, and Spatial-MGCN differed from those in their original versions. SpaGCN was not considered in this experiment due to the unavailability of its embedding.

CCST, and SpaceFlow were messy, and differed from the ground truth. Multiple groups in DeepST were consistent with the ground truth, including clusters 3, 10, 7, 0, 8, and 5, as shown in Fig. 10 C. The ARI of GraphST was 0.63, and its groups 8, 11, 3, 6, 1, 4, 7, and 5 were found in the ground truth as depicted in Fig. 10 F. Further, only STAGATE was able to detect the pyramidal layer in the ground truth, although it could not visualize the five cortex layers as depicted in Fig. 10 E. The UMAP plots of these seven techniques are shown in Supplementary Fig. S5. DeepST was similar to GraphST, and both differed from the other five methods. The above analysis shows that methods of downstream clustering can influence the accuracy of clustering. The performance of the benchmarking algorithms can be improved by selecting an appropriate method of clustering.

3.4. Necessity of PCA and refinement for spatial clustering frameworks

In this section, we examined the impact of PCA-based reduction and methods of refinement on the spatial accuracy of clustering. Table 2 shows that CCST and GraphST used PCA-based reduction on the latent embeddings (Fig. 2 D). Following their predictions of the cluster labels, the results of SpaGCN, DeepST, and GraphST were refined (Fig. 2 F). The KNN-based method of refinement involved selecting the K nearest neighbors, while radius-based refinement involved optimizing the predicted labels of spots within a specified range. The results of previous experiments showed that GraphST, DeepST, and CCST performed well on various datasets. We thus used them to assess the necessity of PCA-based reduction and refinement methods for the frameworks of spatial clustering. The validation data consisted of Stereo-seq-based and Slide-seqV2-enabled datasets of the mouse olfactory bulbs. Because these datasets had no ground truth, the SC score and the DB index were used to evaluate clustering performance.

Based on the implementation of PCA reduction and refinement

methods on the latent embeddings, the clustering scenarios considered here can be classified into six cases as illustrated in Fig. 11 A. Fig. 11 provides a detailed explanation of the SC score in each case for each framework. The cases highlighted in red represent the original choice of each method. The SC score reflects the degree of dispersion of the clusters, and is in the range of $[-1, 1]$. A higher SC score indicates a higher accuracy of clustering. CCST (SC = 0.32) and conST (SC = 0.27) demonstrated good performance on the Stereo-seq-based dataset of the olfactory bulbs of mice. Fig. 12 presents the DB index corresponding to each case. Its range of values was $[0, +\infty]$, and a lower DB index was preferable. Of the seven methods considered, CCST in Case5 (DB = 0.14) and conST in Case4 (DB = 1.14) delivered the best performance. Notably, the default cases for conST, DeepST, GraphST, STAGATE, SpaceFlow, and Spatial-MGCN did not yield optimal results. For example, Case2 was the best scenario for conST (Fig. 12 C) and STAGATE, while CCST and DeepST (Fig. 12 D) recorded their best performance in Case5. The best case for SpaceFlow and Spatial-MGCN is Case3.

Fig. 13 displays the results of identification of spatial domains in the best case for each approach. The manual annotation of these data is shown in Fig. 13 A, where 12 clusters were identified as the ground truth. The spatial regions identified by the methods aligned with their SC scores and DB indices. For instance, the distribution of spots in STAGATE (Fig. 13 E) and GraphST (Fig. 13 F) appeared to be disorganized, and their SC scores were low. Conversely, CCST yielded spatial regions that closely resembled those obtained by manual annotation, as its SC score was the highest and DB index was the lowest. The boundary between groups in CCST was clear, suggesting that selecting a suitable PCA and techniques of refinement can improve the accuracy of clustering.

To strengthen our conclusions, we evaluated these six methods on the Slide-seqV2-based dataset of the olfactory bulbs of mice. SpaGCN and CCST were not considered here because the former required
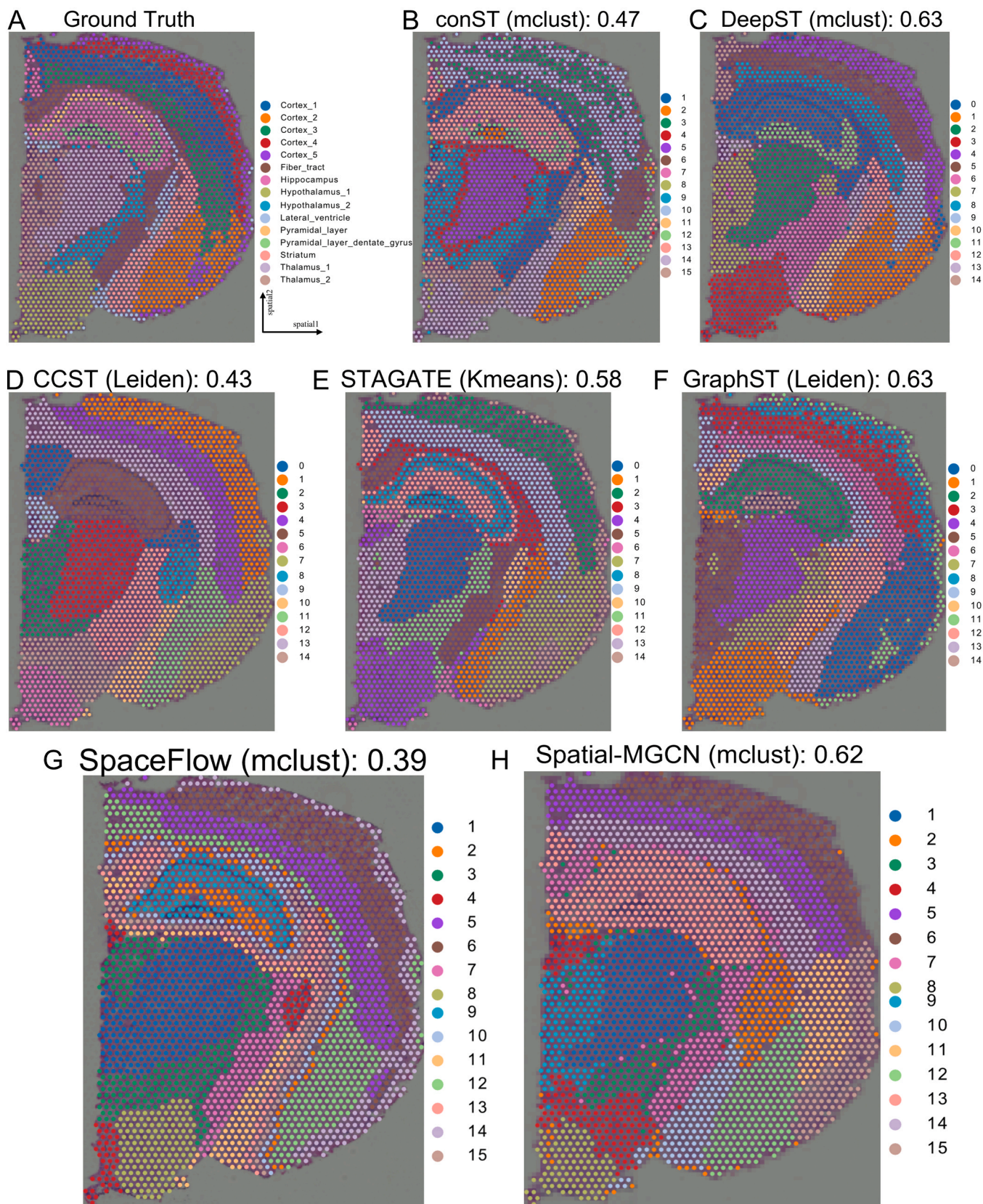
Fig. 10. The results of seven approaches for identifying the spatial domain. The results of clustering depend on the best method of downstream clustering shown in Fig. 9. **A.** The ground truth of the mouse brain dataset as profiled by the Squidpy package. There were 15 groups in this dataset. **B.** The ARI value when using mclust in conST was 0.47. **C, D, E, F, G, H.** The best methods in DeepST, CCST, STAGATE, GraphST, SpaceFlow, and Spatial-MGCN were mclust, Leiden, Kmeans, Leiden, respectively. A highest ARI score of 0.63 was obtained in both DeepST and GraphST. The clusters identified in DeepST and GraphST were similar to the ground truth of the mouse brain dataset.
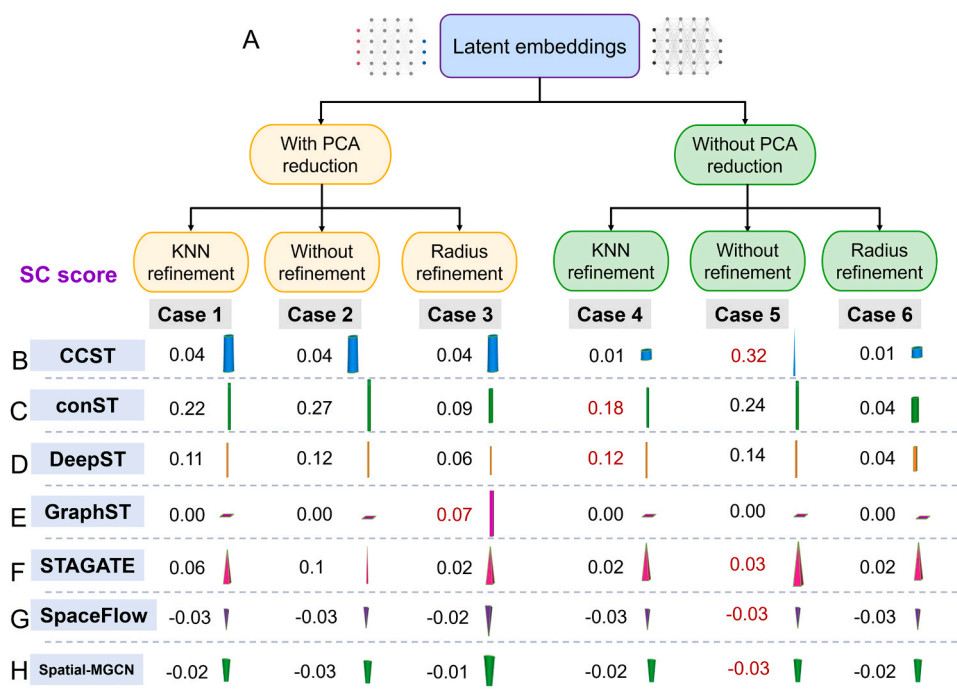
Fig. 11. The impact of PCA-based reduction and refinement methods on clustering performance (SC score). The experimental data were taken from the Stereo-seq-based dataset of the mouse olfactory bulbs. **A.** Six cases of clustering were based on the operations of PCA-based reduction and the refinement of the clustering labels. The accuracy of clustering was evaluated by using the SC score and the DB index because these data had not been manually annotated. **B.** The SC scores for the six cases in CCST. The case given in red represents the original setting. **C.** The original setting of conST was Case4. This indicates that conST did not apply PCA-based reduction and used KNN refinement. The SC score of Case4 was in the middle range for conST. **D, E, F, G, H.** The original PCA-based reduction and refinement operations in DeepST, GraphST, STAGATE, SpaceFlow, and Spatial-MGCN were Case4, Case3, Case5, Case5, and Case5, respectively. It is evident that the primary options in DeepST, STAGATE, SpaceFlow, and Spatial-MGCN were not optimal. SpaGCN was not evaluated in this experiment because histological images were unavailable in this dataset.

histological images and the latter incurred out-of-memory errors. Supplementary Figs S6 and S7 show their SC scores and DB indices, respectively. Interestingly, the original cases did not yield the highest SC score for conST, DeepST, STAGATE, SpaceFlow, and Spatial-MGCN, which is also evident from their DB indices. Consequently, the optimal cases for these six techniques were Case2, Case5, Case3, and Case2, Case3, and Case3, respectively. GraphST delivered the best performance on this dataset, with an SC score of 0.13. The regions detected by each method are presented in Supplementary Fig. S8. DeepST and GraphST performed better than the other four techniques (conST, STAGATE, SpaceFlow, and Spatial-MGCN), with GraphST yielding a clear border between groups. Specifically, GraphST was able to distinguish between clusters 2 and 9 while the other five approaches failed to do. In summary, the default selection of the mode of PCA and the method of refinement was not the most appropriate choice for each method, and selecting different cases (Fig. 11 A) improved their clustering performance.

Furthermore, the eight benchmarking methods are compared on other three datasets (MERFISH, seqFISH, and 10x Xenium, Fig. 3) to estimate their stability. These datasets are MERFISH-based mouse liver data, mouse embryo data from seqFISH, and human breast cancer data from 10x Xenium. The subset of seqFISH-based data is depicted in Supplementary Fig. S9A, wherein 11 annotated clusters are included in this data. It is obvious that GraphST achieved the highest ARI and NMI values (Supplementary Fig. S9C). Hence, the relevant spatial distribution of spots are described in Supplementary Fig. S9B. SpaGCN and CCST are not available here because of the lack of histological image and the out-of-memory error, respectively. Furthermore, these six methods (conST, GraphST, Spatial-MGCN, DeepST, SpaceFlow, and STAGATE) are also compared on MERFISH-based and 10x Xenium-enabled datasets (Supplementary Fig. S10). These methods could be divided into two

levels according to clustering accuracy: GraphST, STAGATE, and Spatial-MGCN were in the first echelon, and the other three methods located in the second echelon (Supplementary Fig. S10C and S10D). These findings align with the outcomes obtained from eight compared approaches applied to the human DLPFC dataset.

3.5. Comparison of runtime and memory usage on multiple datasets of spatial transcriptomics

We also evaluated the computational efficiency of the eight benchmarking methods when applied to the eight spatial transcriptomics datasets. Fig. 14 presents the recorded runtimes and maximum memory usage of each method. The methods were executed by using their original specifications, and the experiments were conducted by using the Python_pyG [74] package and Python. The choices of GNN, methods of downstream clustering, PCA-based reduction, and methods of refinement were the same as before. The evaluation data contains eight various datasets: human brain, mouse brain, breast cancer (10x Visium and 10x Xenium), mouse olfactory bulb (Stereo-seq and Slide-seqV2), mouse liver, and mouse embryo. The Slide-seqV2, Stereo-seq, and 10x Xenium datasets had near-cellular and single-cell resolutions, respectively, that led to a larger number of spots in the dataset of the olfactory bulbs of mice and human breast cancer (Fig. 3). Consequently, the runtime (Fig. 14 A) and memory usage (Fig. 14 B) of the methods on these three datasets were longer and higher. The variations between the Stereo-seq and Slide-seqV2 technologies can be attributed to the number of genes present.

Interestingly, the CCST method recorded the longer runtime and the lower memory usage. However, using one GPU (3090Ti) might have led to an out-of-memory error on the Slide-seqV2 and seqFISH datasets. DeepST and conST required more time and memory due to their
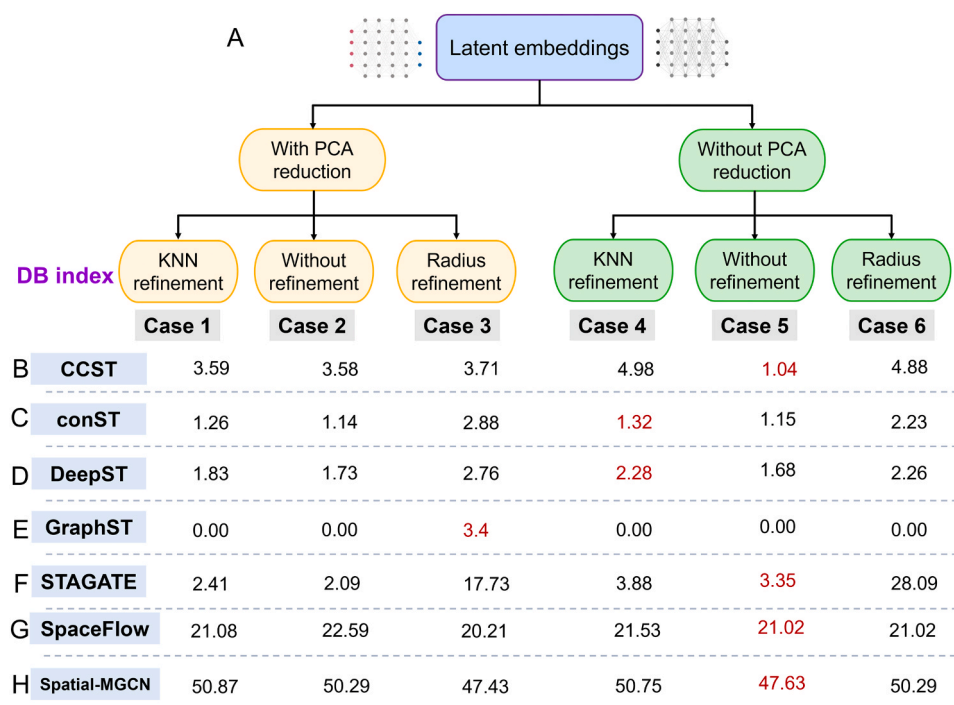
Fig. 12. The DB indices of six cases of clustering by using seven methods. The data used for evaluation were taken from the Stereo-seq-based dataset of the olfactory bulbs of mice. The data had not been manually annotated. **A.** Six cases of clustering were defined based on PCA-based reduction and refinement. **B.** The DB index for each case in CCST, where the case given in red represents the original setting in each technique. The DB index represents the degree of similarity between clusters, and a lower DB index reflects the preferred case. **C.** Case2 involved PCA-based reduction without refinement, had the lowest DB index, and was thus the best case in conST. **D, E, F, G, H.** The DB indices of six clustering cases in DeepST, GraphST, STAGATE, SpaceFlow, and Spatial-MGCN. The optimal choices of PCA-based reduction and refinement can be determined by referring to the definitions of these six cases.

incorporation of multiple neural networks. This increased complexity led to longer runtimes and higher memory usage. GraphST, DeepST, and STAGATE exhibited comparable memory usage, where this can be attributable to the mclust method (Fig. 14 C) that required invoking the R language in Python. SpaGCN was unsuitable for various datasets of ST as it necessitated histological images, and its latent embedding was thus unavailable. Even though both conST and GraphST were constructed by using contrastive learning, the latter outperformed the former on the whole.

In this study, we compared the default settings and optimal choices of eight baseline methods, as depicted in Fig. 14 C and 14 D. The relevant parameters considered here were the choice of GNN, method of downstream clustering, PCA-based reduction, and options for refinement. All of them were found to play a crucial role in determining the quality of the latent embeddings generated by each method, and subsequently impacted the accuracy of clustering (Fig. 2). The experimental results revealed that the original choices of these parameters were not necessarily the best, and the clustering performance of the methods could be significantly enhanced by identifying appropriate alternatives. The identified optimal choices are shown in blue in Fig. 14 D. For instance, the performance of most approaches in terms of identifying distinct spatial domains could be improved by replacing the GNN and downstream clustering methods (Fig. 7 and Fig. 9). The applicability of PCA and suitable options of refinement varied across frameworks, and were dependent on the generated latent representations. This survey provides valuable insights as well as the motivation to develop new frameworks of spatial clustering to improve the performance of prevalent methods.

4. Discussion

In this study, we provided a comprehensive benchmarking of the most advanced frameworks for identifying the spatial domain for spatial

transcriptomics developed within the past two years. This review carefully analyzed and discussed four key factors: the choice of GNN, method of downstream clustering, PCA-based reduction, and method of refinement. Eight benchmarking techniques were categorized into 60 clustering scenarios to determine the optimal parameters for each. These scenarios were thoroughly evaluated and compared across eight datasets of spatial transcriptomics encompassing six technologies: datasets of human DLPFC, human patients of breast cancer, the brains of mice from 10x Visium, the datasets of the olfactory bulbs of mice from Slide-seqV2 and Stereo-seq, mouse liver from MERFISH, mouse embryo from seqFISH, and breast cancer from 10x Xenium. The accuracy of clustering, robustness of various parameters, performance, runtime, and memory usage of the eight frameworks of spatial clustering were meticulously calculated and evaluated based on the experimental results.

The results of this review revealed several intriguing and significant findings. First, GraphST, STAGATE, and Spatial-MGCN were found to be the most accurate methods of spatial clustering among the eight baselines considered here, where this is in line with the results of the original studies on these techniques. Second, using histological images was unnecessary in all the approaches considered (Table 2). In fact, including such image-related information did not necessarily improve the accuracy of clustering (e.g., SpaGCN). Third, the choice of GNN directly influenced the quality of the latent embedding, which subsequently impacted the clustering performance of the methods. The default choices of GNNs in these benchmarking methods may not be optimal. Fourth, a combination of appropriate methods of downstream clustering, PCA-based reduction, and techniques of refinement can enhance the accuracy of clustering (Fig. 14 D). The baselines considered here demonstrated varying performance on the datasets used for their evaluation. Lastly, the numbers of spots and genes in spatial transcriptomics helped determine the runtime and memory usage of the methods. Therefore, when designing a GNN for tasks of spatial clustering (e.g., CCST), the hardware configuration needs to be carefully considered. In
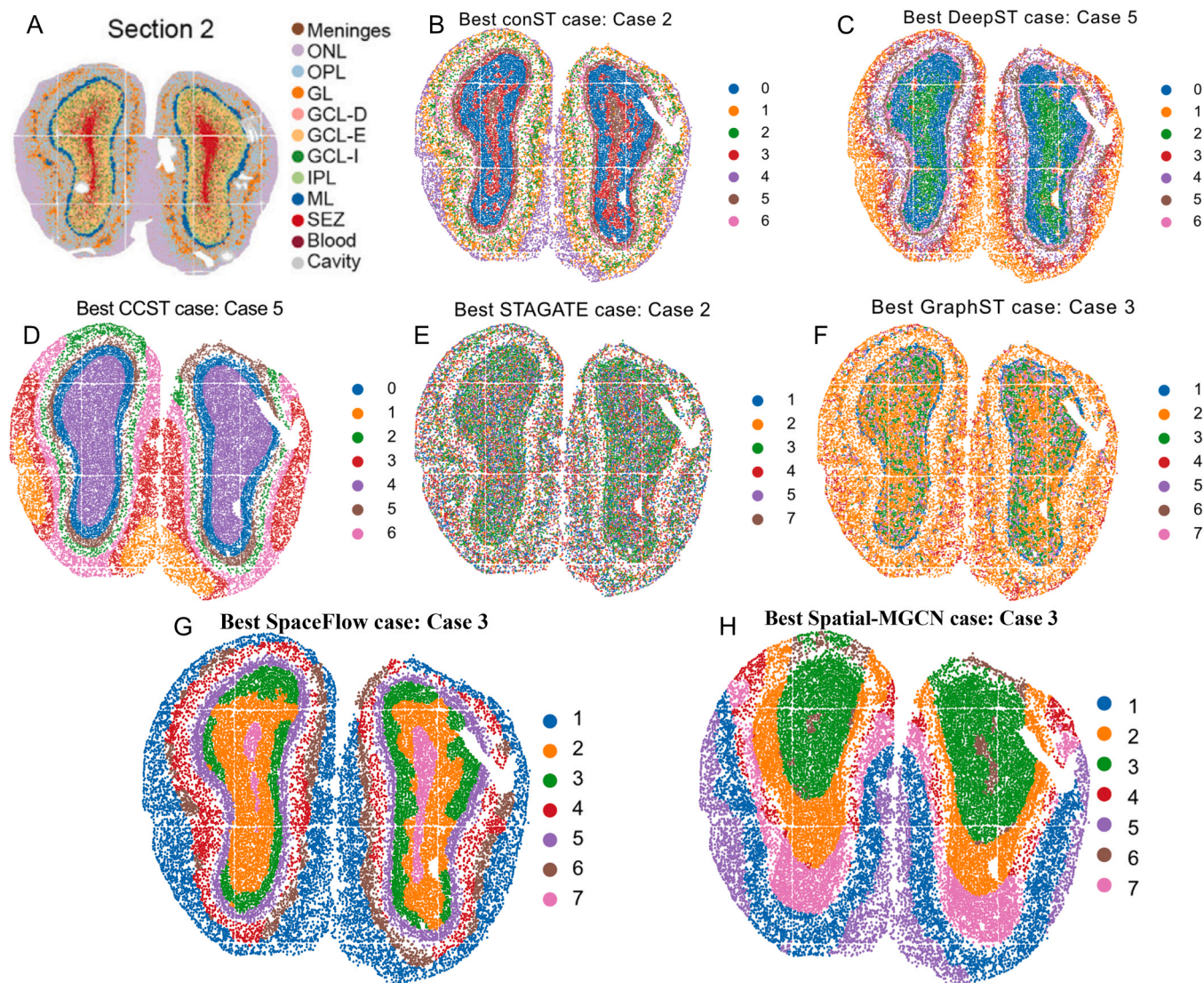
Fig. 13. The results of spatial clustering obtained by using seven approaches on the Stereo-seq-based data on the olfactory bulbs of mice. **A.** The manual annotation for the dataset of the olfactory bulbs of mice was provided by the study on Stereo-seq. There were 12 groups. **B.** The identified clusters in STAGATE according to Case2. Case2 shows the results of applying the PCA without refinement. **C, D, E, F, G, H.** The best cases in CCST, conST, DeepST, GraphST, SpaceFlow, and Spatial-MGCN were Case5, Case2, Case5, Case3, Case3, and Case3, respectively. A higher SC score and a lower DB index were desirable. A comparison with the results of manual annotation shows that CCST had the best clustering performance among the seven baselines, with the highest SC score and the lowest DB index. The borders between the clusters were clearly defined by using CCST. Because no histological image was available for these data, SpaGCN was not considered in this experiment.

conclusion, information on spatial location and histological images in data on ST provide promising avenues for the analysis and better comprehension of ST. Nevertheless, the development of an efficient, generalizable, and scalable framework of spatial clustering with the appropriate parameters remains a challenging area of research.

We envision two promising directions of future research in the field of spatial clustering. First, while unsupervised learning modules are limited in accurately identifying spatial areas of tissues, integrating multiple methods of learning can improve their results of clustering. Such methods include contrastive and semi-supervised methods, generative adversarial neural networks, and graph-based deep learning. Second, a promising avenue of research involves combining single-cell RNA sequencing (scRNA-seq) and the data on ST. scRNA-seq data offer a higher depth of sequencing than spatial transcriptomics, and have been widely used to sequence and annotate various tissues, organs, and cancers. By leveraging the available results of scRNA-seq data for spatial transcriptomics analysis, we can enhance the efficiency and accuracy of clustering. Our future work will primarily focus on exploring

these areas of research to further contribute to the advancement and comprehension of spatial transcriptomics.

CRediT authorship contribution statement

Data availability

The SpaGCN, CCST, STAGATE, DeepST, conST, GraphST, SpaceFlow, and Spatial-MGCN benchmarking methods were discussed in this review. The codes for these frameworks of spatial clustering are available as follows: 1) SpaGCN: https://github.com/jianhuupenn/SpaGCN; 2)
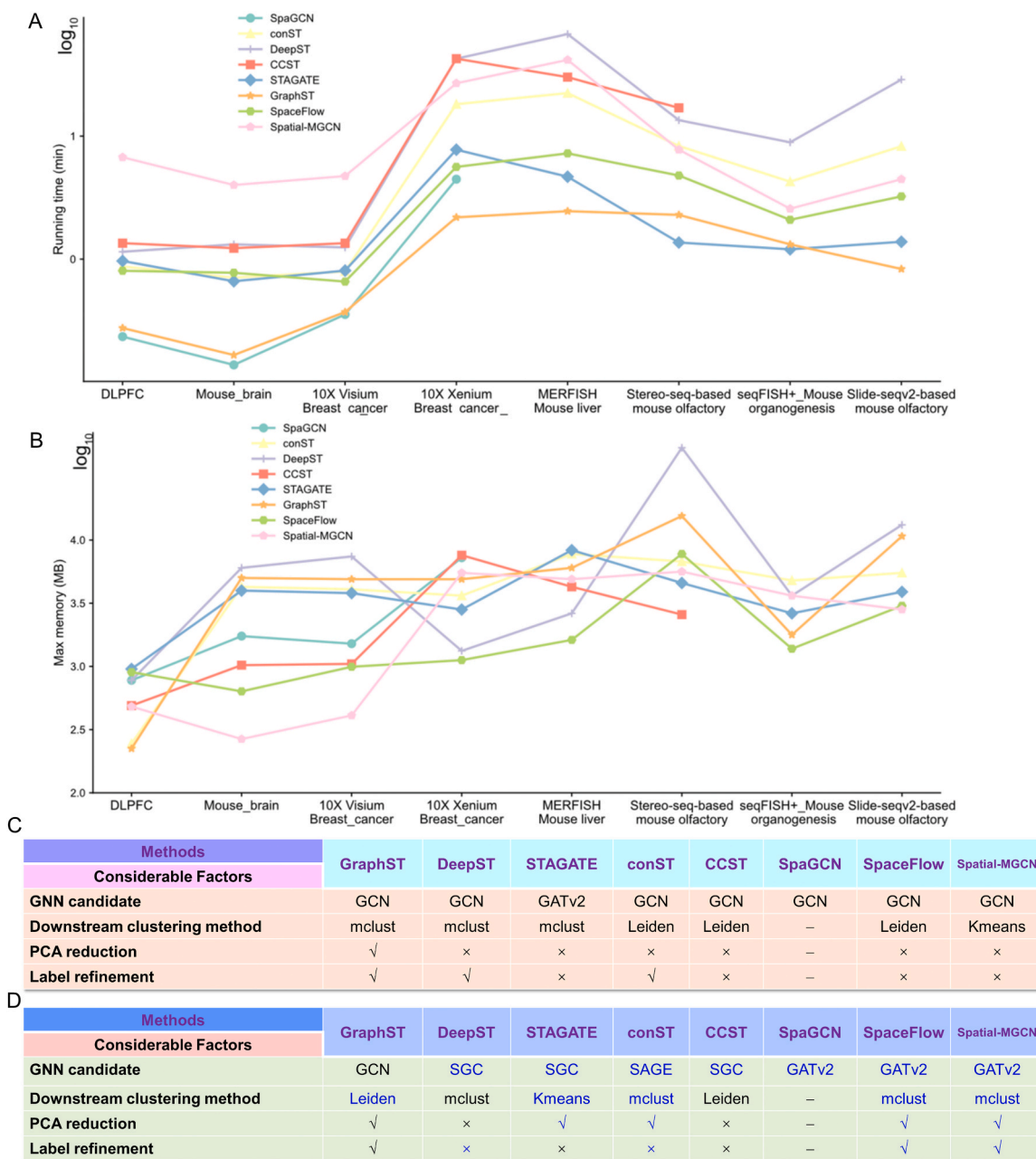
Fig. 14. The runtime and memory usage of eight benchmarking methods for spatial transcriptomics analysis. **A.** The runtimes of the eight benchmarking techniques when they were applied to eight datasets of spatial transcriptomics. The datasets were the human brain, mouse brain, human breast cancer, mouse liver, mouse embryo, Stereo-seq-based, and Slide-seq-based datasets of the olfactory bulbs of mice (Fig. 3). **B.** The memory usage of the eight baselines on the eight datasets of ST. The embedding of SpaGCN was unavailable, and histological images were necessary for it. Hence, SpaGCN was tested only on the human brain, mouse brain, and human breast cancer datasets. CCST would lead to out-of-memory error on seqFISH and Slide-seqV2 datasets. **C.** The default settings for GNN selection, methods of downstream clustering, PCA-based reduction, and methods of refinement in the eight frameworks of spatial clustering. The implementations of the PCA and refinement are indicated by √ and ×, respectively. **D.** The improved settings for each method based on the experiment results of this survey. The parameters that differed from the original settings in each method are highlighted by blue words.

CCST: https://github.com/xiaoyeye/CCST; 3) STAGATE: https://github.com/zhanglabtools/STAGATE; 4) DeepST: https://github.com/JiangBioLab/DeepST; 5) conST: https://github.com/ys-zong/conST; 6) GraphST: https://github.com/JinmiaoChenLab/GraphST; 7) SpaceFlow: https://github.com/hongleir/SpaceFlow; 8) Spatial-MGCN: https://github.com/cs-wangbo/Spatial-MGCN.Eight datasets of spatial transcriptomics were used in this survey to evaluate the above baselines. We have provided the links to the original and the pre-processed resources for each. 1) Human DLPFC: The primary source: https://www.nature.com/articles/s41593-020-00787-0; the pre-processed source:

https://github.com/LieberInstitute/spatialLIBD. 2) Human breast cancer: The primary source: https://www.10xgenomics.com/resources/datasets/human-breast-cancer-block-a-section-1-1-standard-1-1-0; the pre-processed source: https://github.com/JinmiaoChenLab/SEDR_analyses/. 3) Mouse brain: The primary source: https://support.10xgenomics.com/spatial-gene-expression/datasets; the pre-processed source: https://github.com/JinmiaoChenLab/SEDR_analyses/. 4) Data on the olfactory bulbs of mice for Slide-seqV2: The primary source: https://singlecell.broadinstitute.org/single_cell/study/SCP815/highly-sensitive-spatial-transcriptomics-at-near-cellular-

resolution-with-slide-seqv2#study-summary; the processed version: https://stagate.readthedocs.io/en/latest/T3_Slide-seqV2.html. 5) Data on the olfactory bulbs of mice for Stereo-seq: The primary source: https://github.com/JinmiaoChenLab/SEDR_analyses/tree/master/ data; the processed version: https://stagate.readthedocs.io/en/latest/ T4_Stereo.html. 6) Mouse liver from MERFISH: the primary source: https://info.vizgen.com/mouse-liver-access; the processed version: https://squidpy.readthedocs.io/en/stable/notebooks/tutorials/ tutorial_vizgen_mouse_liver.html#single-cell-clustering-of-vizgen-merfish-mouse-liver-data; 7) Mouse embryo from seqFISH: the primary source: https://crukci.shinyapps.io/SpatialMouseAtlas/; the processed version: https://squidpy.readthedocs.io/en/stable/notebooks/tuto-rials/tutorial_seqfish.html; 8) Human breasr cancer from 10x Xenium: the primary source: https://www.10xgenomics.com/products/xenium-in-situ/preview-dataset-human-breast; https://squidpy.readthedocs.io/ en/stable/notebooks/tutorials/tutorial_xenium.html.The codes for this review have been summarized at https://github.com/narutoten520/ Benchmark_SRT.

Acknowledgments

Authors' contributions

MY and ML designed this study, and TL and ZF prepared the codes for analyzing the benchmarking methods. ZF designed the relevant workflow for assessing factors crucial to the clustering frameworks, and TL wrote the entire paper. ML checked the experimental results and provided revised ideas for them, and MY determined the process of improving the analytical logic. YY supplemented the experimental results in the revised version, incuding adding the methods and datasets. TL and ZZ revised the first submission, including the figures and text. All authors participated in revising the manuscript, and have read and approved the final manuscript.

Competing interests

All of the authors have no conflicts of interest to declare.

Code availability

In addition to comparing the original GNN-based algorithms of spatial clustering, this review discussed the effects of several factors on the performance of cell-type identification for data on ST. The relevant codes are available at https://github.com/narutoten520/Bench mark_SRT.

Appendix A. Supporting information

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.csbj.2023.11.055.

References

[1] Rao A, Barkley D, França GS, Yanai I. Exploring tissue architecture using spatial transcriptomics. Nature 2021;596:211–20. https://doi.org/10.1038/s41586-021-03634-9.

[2] Williams CG, Lee HJ, Asatsuma T, Vento-Tormo R, Haque A. An introduction to spatial transcriptomics for biomedical research. Genome Med 2022;14:68. https://doi.org/10.1186/s13073-022-01075-1.

[3] Zhang M, Eichhorn SW, Zingg B, Yao Z, Cotter K, Zeng H, et al. Spatially resolved cell atlas of the mouse primary motor cortex by MERFISH. Nature 2021;598:137–43. https://doi.org/10.1038/s41586-021-03705-x.

[4] Shah S, Takei Y, Zhou W, Lubeck E, Yun J, Eng C-HL, et al. Dynamics and spatial genomics of the nascent transcriptome by intron seqFISH. e16 Cell 2018;174:363–76. https://doi.org/10.1016/j.cell.2018.05.035.

[5] Codeluppi S, Borm LE, Zeisel A, La Manno G, Van Lunteren JA, Svensson CI, et al. Spatial organization of the somatosensory cortex revealed by osmFISH. Nat Methods 2018;15:932–5. https://doi.org/10.1038/s41592-018-0175-z.

[6] Wang X, He Y, Zhang Q, Ren X, Zhang Z. Direct comparative analyses of 10X genomics chromium and smart-seq2. Genom Proteom Bioinforma 2021;19:253–66. https://doi.org/10.1016/j.gpb.2020.02.005.

[7] Stickels RR, Murray E, Kumar P, Li J, Marshall JL, Di Bella DJ, et al. Highly sensitive spatial transcriptomics at near-cellular resolution with slide-seqV2. Nat Biotechnol 2021;39:313–9. https://doi.org/10.1038/s41587-020-0739-1.

[8] Chen A, Liao S, Cheng M, Ma K, Wu L, Lai Y, et al. Spatiotemporal transcriptomic atlas of mouse organogenesis using DNA nanoball-patterned arrays. e21 Cell 2022;185:1777–92. https://doi.org/10.1016/j.cell.2022.04.003.

[9] Palla G, Fischer DS, Regev A, Theis FJ. Spatial components of molecular tissue biology. Nat Biotechnol 2022;40:308–18. https://doi.org/10.1038/s41587-021-01182-1.

[10] Xu Z, Wang W, Yang T, Chen J, Huang Y, Gould J, et al. STOmicsDB: a database of spatial transcriptomic data. Genomics 2022. https://doi.org/10.1101/2022.03.11.481421.

[11] Teves J.M., Won K.J. Mapping Cellular Coordinates through Advances in Spatial Transcriptomics Technology n.d.

[12] Wang N, Li X, Wang R, Ding Z. Spatial transcriptomics and proteomics technologies for deconvoluting the tumor microenvironment. Biotechnol J 2021;16:2100041. https://doi.org/10.1002/biot.202100041.

[13] Jiang R, Li Z, Jia Y, Li S, Chen S. SINFONIA: scalable identification of spatially variable genes for deciphering spatial domains. Cells 2023;12:604. https://doi.org/10.3390/cells12040604.

[14] Tang Z, Zhang T, Yang B, Su J, Song Q. spaCI: deciphering spatial cellular communications through adaptive graph model. bbac563 Brief Bioinforma 2023;24. https://doi.org/10.1093/bib/bbac563.

[15] Liu W, Liao X, Yang Y, Lin H, Yeong J, Zhou X, et al. Joint dimension reduction and clustering analysis of single-cell RNA-seq and spatial transcriptomics data. Nucleic Acids Res 2022;50:e72. https://doi.org/10.1093/nar/gkac219.

[16] Bu Z, Li H-J, Zhang C, Cao J, Li A, Shi Y. Graph K-means based on leader identification, dynamic game, and opinion dynamics. IEEE Trans Knowl Data Eng 2020;32:1348–61. https://doi.org/10.1109/TKDE.2019.2903712.

[17] Cheng A, Hu G, Li WV. Benchmarking cell-type clustering methods for spatially resolved transcriptomics data. bbac475 Brief Bioinforma 2023;24. https://doi.org/10.1093/bib/bbac475.

[18] Avesani S., Viesi E., Alessandrì L., Motterle G., Bonnici V., Beccuti M., et al. Stardust: improving spatial transcriptomics data analysis through space-aware modularity optimization-based clustering n.d.

[19] Dries R, Zhu Q, Dong R, Eng C-HL, Li H, Liu K, et al. Giotto: a toolbox for integrative analysis and visualization of spatial expression data. Genome Biol 2021;22:78. https://doi.org/10.1186/s13059-021-02286-2.

[20] Pham D, Tan X, Xu J, Grice LF, Lam PY, Raghubar A, et al. stLearn: integrating spatial location, tissue morphology and gene expression to find cell types, cell-cell interactions and spatial trajectories within undissociated tissues. Bioinformatics 2020. https://doi.org/10.1101/2020.05.31.125658.

[21] Tan X, Su A, Tran M, Nguyen Q. SpaCell: integrating tissue morphology and spatial gene expression to predict disease cells. Bioinformatics 2020;36:2293–4. https://doi.org/10.1093/bioinformatics/btz914.

[22] Zhao E, Stone MR, Ren X, Guenthoer J, Smythe KS, Pulliam T, et al. Spatial transcriptomics at subspot resolution with BayesSpace. Nat Biotechnol 2021;39:1375–84. https://doi.org/10.1038/s41587-021-00935-2.

[23] Yang Y, Shi X, Liu W, Zhou Q, Chan Lau M, Chun Tatt Lim J, et al. SC-MEB: spatial clustering with hidden Markov random field using empirical Bayes. bbab466 Brief Bioinforma 2022;23. https://doi.org/10.1093/bib/bbab466.

[24] Li Z, Song T, Yong J, Kuang R. Imputation of spatially-resolved transcriptomes by graph-regularized tensor completion. PLoS Comput Biol 2021;17:e1008218. https://doi.org/10.1371/journal.pcbi.1008218.

[25] Liu Y, Wang T, Duggan B, Sharpnack M, Huang K, Zhang J, et al. SPCS: a spatial and pattern combined smoothing method for spatial transcriptomic expression. bbac116 Brief Bioinforma 2022;23. https://doi.org/10.1093/bib/bbac116.

[26] Wu Z, Pan S, Chen F, Long G, Zhang C, Yu PS. A comprehensive survey on graph neural networks. IEEE Trans Neural Netw Learn Syst 2021;32:4–24. https://doi.org/10.1109/TNNLS.2020.2978386.

[27] Hu J, Li X, Coleman K, Schroeder A, Ma N, Irwin DJ, et al. SpaGCN: integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. Nat Methods 2021;18:1342–51. https://doi.org/10.1038/s41592-021-01255-8.

[28] Fu H., Xu H., Chong K., Li M., Ang K.S., Lee H.K., et al. 1 Unsupervised Spatially Embedded Deep Representation of Spatial 2 Transcriptomics n.d.

[29] Li J, Chen S, Pan X, Yuan Y, Shen H-B. Cell clustering for spatial transcriptomics data with graph neural networks. Nat Comput Sci 2022;2:399–408. https://doi.org/10.1038/s43588-022-00266-5.

[30] Zong Y, Yu T, Wang X, Wang Y, Hu Z, Li Y. conST: an interpretable multi-modal contrastive learning framework for spatial transcriptomics. Bioinformatics 2022. https://doi.org/10.1101/2022.01.14.476408.

[31] Dong K, Zhang S. Deciphering spatial domains from spatially resolved transcriptomics with an adaptive graph attention auto-encoder. Nat Commun 2022;13:1739. https://doi.org/10.1038/s41467-022-29439-6.

[32] Xu C, Jin X, Wei S, Wang P, Luo M, Xu Z, et al. DeepST: identifying spatial domains in spatial transcriptomics by deep learning. Nucleic Acids Res 2022;50:e131. https://doi.org/10.1093/nar/gkac901.

[33] Long Y, Ang KS, Li M, Chong KLK, Sethi R, Zhong C, et al. Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST. Nat Commun 2023;14:1155. https://doi.org/10.1038/s41467-023-36796-3.

[34] Zeng Z, Li Y, Li Y, Luo Y. Statistical and machine learning methods for spatially resolved transcriptomics data analysis. Genome Biol 2022;23:83. https://doi.org/10.1186/s13059-022-02653-7.

[35] Ezugwu AE, Ikotun AM, Oyelade OO, Abualigah L, Agushaka JO, Eke CI, et al. A comprehensive survey of clustering algorithms: state-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. Eng Appl Artif Intell 2022;110:104743. https://doi.org/10.1016/j.engappai.2022.104743.

[36] Zhou S., Xu H., Zheng Z., Chen J., li Z., Bu J., et al. A Comprehensive Survey on Deep Clustering: Taxonomy, Challenges, and Future Directions 2022.

[37] Li Y, Stanojevic S, Garmire LX. Emerging artificial intelligence applications in spatial transcriptomics analysis. Comput Struct Biotechnol J 2022;20:2895–908. https://doi.org/10.1016/j.csbj.2022.05.056.

[38] Fang S, Chen B, Zhang Y, Sun H, Liu L, Liu S, et al. Computational approaches and challenges in spatial transcriptomics. Genom Proteom Bioinforma 2023;21:24–47. https://doi.org/10.1016/j.gpb.2022.10.001.

[39] Satija R, Farrell JA, Gennert D, Schier AF, Regev A. Spatial reconstruction of single-cell gene expression data. Nat Biotechnol 2015;33:495–502. https://doi.org/10.1038/nbt.3192.

[40] Xu D, Tian Y. A comprehensive survey of clustering algorithms. Ann Data Sci 2015;2:165–93. https://doi.org/10.1007/s40745-015-0040-1.

[41] Hu J, Schroeder A, Coleman K, Chen C, Auerbach BJ, Li M. Statistical and machine learning methods for spatially resolved transcriptomics with histology. Comput Struct Biotechnol J 2021;19:3829–41. https://doi.org/10.1016/j.csbj.2021.06.052.

[42] Ren H, Walker BL, Cang Z, Nie Q. Identifying multicellular spatiotemporal organization of cells with SpaceFlow. Nat Commun 2022;13:4076. https://doi.org/10.1038/s41467-022-31739-w.

[43] Wang B, Luo J, Liu Y, Shi W, Xiong Z, Shen C, et al. Spatial-MGCN: a novel multi-view graph convolutional network for identifying spatial domains with attention mechanism. bbad262 Brief Bioinforma 2023;24. https://doi.org/10.1093/bib/bbad262.

[44] Zhang S, Tong H, Xu J, Maciejewski R. Graph convolutional networks: a comprehensive review. Comput Soc Netw 2019;6:11. https://doi.org/10.1186/s40649-019-0069-y.

[45] Bullard JH, Purdom E, Hansen KD, Dudoit S. Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments. BMC Bioinforma 2010;11:94. https://doi.org/10.1186/1471-2105-11-94.

[46] Ung K, Huang T-W, Lozzi B, Woo J, Hanson E, Pekarek B, et al. Olfactory bulb astrocytes mediate sensory circuit processing through Sox9 in the mouse brain. Nat Commun 2021;12:5230. https://doi.org/10.1038/s41467-021-25444-3.

[47] Maynard KR, Collado-Torres L, Weber LM, Uytingco C, Barry BK, Williams SR, et al. Transcriptome-scale spatial gene expression in the human dorsolateral prefrontal cortex. Nat Neurosci 2021;24:425–36. https://doi.org/10.1038/s41593-020-00787-0.

[48] Yip SH, Romanò N, Gustafson P, Hodson DJ, Williams EJ, Kokay IC, et al. Elevated Prolactin during pregnancy drives a phenotypic switch in mouse hypothalamic Dopaminergic neurons. e5 Cell Rep 2019;26:1787–99. https://doi.org/10.1016/j.celrep.2019.01.067.

[49] Svensson V, Teichmann SA, Stegle O. SpatialDE: identification of spatially variable genes. Nat Methods 2018;15:343–6. https://doi.org/10.1038/nmeth.4636.

[50] Zhu J, Sun S, Zhou X. SPARK-X: non-parametric modeling enables scalable and robust detection of spatial expression patterns for large spatial transcriptomic studies. Genome Biol 2021;22:184. https://doi.org/10.1186/s13059-021-02404-0.

[51] Park C, Han J, Yu H. Deep multiplex graph infomax: attentive multiplex network embedding using global information. Knowl-Based Syst 2020;197:105861. https://doi.org/10.1016/j.knosys.2020.105861.

[52] Chen Y, Zhou S, Li M, Zhao F, Qi J. STEEL enables high-resolution delineation of spatiotemporal transcriptomic data. bbad068 Brief Bioinforma 2023;24. https://doi.org/10.1093/bib/bbad068.

[53] Fraley C., Raftery A.E. MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering* n.d.

[54] Ratz M, Von Berlin L, Larsson L, Martin M, Westholm JO, La Manno G, et al. Clonal relations in the mouse brain revealed by single-cell and spatial transcriptomics. Nat Neurosci 2022;25:285–94. https://doi.org/10.1038/s41593-022-01011-x.

[55] Shao X, Li C, Yang H, Lu X, Liao J, Qian J, et al. Knowledge-graph-based cell-cell communication inference for spatially resolved transcriptomic data with SpaTalk. Nat Commun 2022;13:4429. https://doi.org/10.1038/s41467-022-32111-8.

[56] Wolf FA, Angerer P, Theis FJ. SCANPY: large-scale single-cell gene expression data analysis. Genome Biol 2018;19:15. https://doi.org/10.1186/s13059-017-1382-0.

[57] Kipf T.N., Welling M. Variational Graph Auto-Encoders, 2016.

[58] Zhang Z. Introduction to machine learning: k-nearest neighbors. Ann Transl Med 2016;4:218. https://doi.org/10.21037/atm.2016.03.37.

[59] Cao S., Xu P., Clifton D.A. How to Understand Masked Autoencoders, 2022.

[60] Jaiswal A, Babu AR, Zadeh MZ, Banerjee D, Makedon F. A survey on contrastive self-supervised learning. Technologies 2020;9:2. https://doi.org/10.3390/technologies9010002.

[61] Miller BF, Bambah-Mukku D, Dulac C, Zhuang X, Fan J. Characterizing spatial gene expression heterogeneity in spatially resolved single-cell transcriptomic data with nonuniform cellular densities. Genome Res 2021;31:1843–55. https://doi.org/10.1101/gr.271288.120.

[62] Yu Z, Lu Y, Wang Y, Tang F, Wong K-C, Li X. ZINB-based graph embedding autoencoder for single-cell RNA-Seq interpretations. AAAI 2022;36:4671–9. https://doi.org/10.1609/aaai.v36i4.20392.

[63] Pardo B, Spangler A, Weber LM, Hicks SC, Jaffe AE, Martinowich K, et al. spatialLIBD: an R/Bioconductor package to visualize spatially-resolved transcriptomics data. Bioinformatics 2021. https://doi.org/10.1101/2021.04.29.440149.

[64] Hao M, Hua K, Zhang X. SOMDE: a scalable method for identifying spatially variable genes with self-organizing map. Bioinformatics 2021;37:4392–8. https://doi.org/10.1093/bioinformatics/btab471.

[65] Flati T, Gioiosa S, Chillemi G, Mele A, Oliverio A, Mannironi C, et al. A gene expression atlas for different kinds of stress in the mouse brain. Sci Data 2020;7:437. https://doi.org/10.1038/s41597-020-00772-z.

[66] Palla G, Spitzer H, Klein M, Fischer D, Schaar AC, Kuemmerle LB, et al. Squidpy: a scalable framework for spatial omics analysis. Nat Methods 2022;19:171–8. https://doi.org/10.1038/s41592-021-01358-2.

[67] Khodosevich K, Lazarini F, von Engelhardt J, Kaneko H, Lledo P-M, Monyer H. Connective tissue growth factor regulates interneuron survival and information processing in the olfactory bulb. Neuron 2013;79:1136–51. https://doi.org/10.1016/j.neuron.2013.07.011.

[68] Kuan L, Li Y, Lau C, Feng D, Bernard A, Sunkin SM, et al. Neuroinformatics of the allen mouse brain connectivity atlas. Methods 2015;73:4–17. https://doi.org/10.1016/j.ymeth.2014.12.013.

[69] Janesick A, Shelansky R, Gottscho AD, Wagner F, Rouault M, Beliakoff G, et al. High resolution mapping of the breast cancer tumor microenvironment using integrated single cell, spatial and in situ analysis of FFPE tissue. Cancer Biol 2022. https://doi.org/10.1101/2022.10.06.510405.

[70] Lohoff T, Ghazanfar S, Missarova A, Koulena N, Pierson N, Griffiths JA, et al. Highly multiplexed spatially resolved gene expression profiling of mouse organogenesis. Dev Biol 2020. https://doi.org/10.1101/2020.11.20.391896.

[71] Xia C, Fan J, Emanuel G, Hao J, Zhuang X. Spatial transcriptome profiling by MERFISH reveals subcellular RNA compartmentalization and cell cycle-dependent gene expression. Proc Natl Acad Sci USA 2019;116:19490–9. https://doi.org/10.1073/pnas.1912459116.

[72] Zeng H, Shen EH, Hohmann JG, Oh SW, Bernard A, Royall JJ, et al. Large-scale cellular-resolution gene profiling in human neocortex reveals species-specific molecular signatures. Cell 2012;149:483–96. https://doi.org/10.1016/j.cell.2012.02.052.

[73] Zeng Y, Wei Z, Yu W, Yin R, Yuan Y, Li B, et al. Spatial transcriptomics prediction from histology jointly through transformer and graph neural networks. bbac297 Brief Bioinforma 2022;23. https://doi.org/10.1093/bib/bbac297.

[74] Fey M., Lenssen J.E. Fast Graph Representation Learning with PyTorch Geometric, 2019.

[75] Ghosh S., Halappanavar M., Tumeo A., Kalyanaraman A., Lu H., Chavarria-Miranda D., et al. Distributed Louvain Algorithm for Graph Community Detection. 2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS), Vancouver, BC: IEEE; 2018, p. 885–95. ⟨https://doi.org/10.1109/IPDPS.2018.00098⟩.

[76] Kim T., Oh J., Kim N., Cho S., Yun S.-Y. Comparing Kullback-Leibler Divergence and Mean Squared Error Loss in Knowledge Distillation, 2021.

[77] Clevert D.-A., Unterthiner T., Hochreiter S. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs), 2016.

[78] Cao Y, Wang H, Zhao W, Duan B, Zhang X. A new method to construct the KD tree based on presorted results. Complexity 2020;2020:1–7. https://doi.org/10.1155/2020/8883945.

[79] Agarap A.F. Deep Learning using Rectified Linear Units (ReLU), 2019.

[80] Zhang S, Wong H-S, Shen Y. Generalized adjusted rand indices for cluster ensembles. Pattern Recognit 2012;45:2214–26. https://doi.org/10.1016/j.patcog.2011.11.017.

[81] Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., et al. Scikit-learn: Machine Learning in Python. MACHINE LEARNING IN PYTHON n.d.

[82] Addagarla SK, Amalanathan A. Probabilistic unsupervised machine learning approach for a similar image recommender system for E-commerce. Symmetry 2020;12:1783. https://doi.org/10.3390/sym12111783.

[83] Alessandrì L, Cordero F, Beccuti M, Arigoni M, Olivero M, Romano G, et al. rCASC: reproducible classification analysis of single-cell sequencing data. giz105 GigaScience 2019;8. https://doi.org/10.1093/gigascience/giz105.

[84] McInnes L., Healy J., Melville J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction, 2020.

[85] Brody S., Alon U., Yahav E. How Attentive are Graph Attention Networks?, 2022.

[86] Wu F., Zhang T., Souza de A.H. Jr, Fifty C., Yu T., Weinberger K.Q. Simplifying Graph Convolutional Networks 2019.

[87] Du J., Zhang S., Wu G., Moura J.M.F., Kar S. Topology Adaptive Graph Convolutional Networks, 2018.

[88] Hamilton W.L., Ying R., Leskovec J. Inductive Representation Learning on Large Graphs, 2018.

[89] Scrucca L, Fop M, Murphy TBrendan, Raftery AE. Mclust 5: clustering, classification and density estimation using gaussian finite mixture models. R J 2016;8:289. https://doi.org/10.32614/RJ-2016-021.